

**SISTEM DETEKSI PERHATIAN OPERATOR KAMERA
PENGAWAS TERHADAP MONITOR MENGGUNAKAN HAAR
CASCADE CLASSIFIER**

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Teknik

Disusun oleh:
Devi Ayu Ratnasari
NIM: 145150301111038



PROGRAM STUDI TEKNIK KOMPUTER
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2018

PENGESAHAN

SISTEM DETEKSI PERHATIAN OPERATOR KAMERA PENGAWAS TERHADAP
MONITOR MENGGUNAKAN HAAR CASCADE CLASSIFIER

SKRIPSI

KEMINATAN TEKNIK KOMPUTER

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Teknik

Disusun Oleh :
Devi Ayu Ratnasari
NIM: 145150301111038

Skripsi ini telah diuji dan dinyatakan lulus pada
30 Juli 2018

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Hurriyatul Fitriyah, S.T, M.Sc
NIP. 19851001 201504 2 003

Mochammad Hannats Hanafi Ichsan, S.ST, M.T
NIK. 201405 881229 1 001

Mengetahui
Ketua Jurusan Teknik Informatika

Tri Astoto Kurniawan, S.T., M.T., Ph.D
NIP: 19710518 200312 1 001

PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 30 Juli 2018

Devi Ayu Ratnasari

NIM: 145150301111038



KATA PENGANTAR

Puji syukur penulis ucapkan kepada kehadiran Tuhan Yang Maha Esa, yang telah melimpahkan rahmat, berkah, dan karunia-Nya maka penulis dapat menyelesaikan penelitian dan penyusunan laporan skripsi dengan judul “Sistem Deteksi Perhatian Operator Kamera Pengawas Terhadap Monitor Menggunakan Haar Cascade Clasifier”.

Penulis menyadari bahwa selama proses pelaksanaan penelitian hingga penulisan laporan skripsi ini penulis mendapat bantuan dari berbagai pihak, oleh karena itu penulis hendak menyampaikan rasa hormat dan ungkapan terimakasih kepada:

1. Ibu Hurriyatul Fitriyah, S.T, M.Sc dan Bapak Mochammad Hannats Hanafi Ichsan, S.ST, M.T selaku dosen pembimbing yang dengan sabar meluangkan waktu untuk membimbing dan memberi arahan pada penulis sehingga penelitian ini dapat terselesaikan
2. Bapak Tri Astoto Kurniawan, S.T., M.T., Ph.D selaku Ketua Jurusan Teknik Informatika.
3. Kedua orang tua yang sudah memberikan berbagai bentuk dukungan kepada peneliti baik dengan do’a maupun dukungan moral.
4. Rekan skripsi Ganda Wibawa Putra, Octavian Metta Wisnu serta rekan-rekan penulis yang telah membantu dan membimbing selama masa perkuliahan hingga pengerjaan skripsi dengan saran maupun motivasi
5. Seluruh civitas akademik di Fakultas Ilmu Komputer yang telah membantu peneliti selama menempuh studi di Universitas Brawijaya Malang

Tidak dapat dipungkiri bahwa dalam pengerjaan penelitian dan penyusunan laporan ini belum sempurna, maka penulis mengharap kritik dan saran yang membangun. Demikian, semoga penelitian ini bermanfaat bagi pembacanya dan pihak yang menggunakannya.

Malang, 30 Juli 2018

Penulis

Devi8ayu8ratnasari95@gmail.com

ABSTRAK

Operator pengawas kamera diwajibkan melakukan pengawasan monitor dengan durasi shift rata-rata 6 jam sampai 8 jam yang menyebabkan menurunnya tingkat perhatian dalam mengawasi monitor sehingga tugas pengawasannya kurang maksimal. Karena tidak adanya pengawas kinerja operator untuk mengetahui perhatian operator terhadap monitor, maka dibuatlah alat pengawas operator secara realtime. Sistem ini menggunakan kamera sebagai sumber masukan, Raspberry Pi sebagai pemroses dan buzzer sebagai keluaran sistem. Penelitian ini menggunakan konversi warna HSV untuk menentukan piksel yang mengandung warna kulit dan *thresholding* untuk merubah gambar menjadi biner (hitam atau putih). Posisi duduk dan berdiri ditentukan dengan membagi frame menjadi dua yaitu atas dan bawah lalu dilakukan perhitungan piksel putih pada setiap frame. Jika piksel putih pada frame bawah lebih banyak daripada frame atas maka akan disimpulkan subjek dalam posisi duduk dan sebaliknya untuk posisi berdiri. Jika subjek terdeteksi duduk, maka selanjutnya adalah deteksi wajah dan mata dengan algoritma *Haar Cascade Classifier*. Jika wajah dan mata subjek terdeteksi, maka subjek dikategorikan dalam kondisi memperhatikan monitor, dan jika tidak terdeteksi maka subjek dikategorikan tidak memperhatikan monitor. Buzzer sebagai hasil keluaran sistem akan berbunyi jika subjek dalam kondisi tidak memperhatikan monitor dan buzzer akan diam jika subjek dalam kondisi memperhatikan monitor. Hasil pengujian dari sistem ini berupa deteksi duduk dan berdiri dengan tingkat akurasi 100%. Hasil deteksi wajah dan mata untuk klasifikasi perhatian atau tidak memperhatikan monitor didapat tingkat akurasi sebesar 98.33%.

Kata kunci: Deteksi Perhatian, Operator Pengawas Kamera, Haar cascade classifier, Raspberry Pi

ABSTRACT

Surveillance camera operators have to watching the monitor with an average shift duration 6 hours to 8 hours which caused the level of focus in watching the monitor decrease so the surveillance task is insufficient. There is no supervisor for the performance of the operators to know the operator attention to monitors, then the surveillance tool for operator in realtime has been made. The system uses the camera as a source of input, Raspberry Pi as the processor and bell as the output of the system. This research use the HSV color conversion to determine which pixels containing skin color and thresholding to change the picture to a binary image (black or white). Sitting and standing position is determined by dividing the frame into the upper and lower then do the calculations of white pixels in each frame. If the white pixels in the frame below a lot more than top frame it will the categorized in a sitting position, instead for the standing position. If any body is detected, then do the face and eyes detection with Haar Cascade Classifier algorithm. If the face and eyes of the body are detected, then the subject categorized in focus condition, and if it is not detected then the subject categorized in didn't focus. Bel as a result of the system, the bel will beep if the subject categorized didn't focus and Bell will be silent if the subject categorized focus. The results of the testing of this system in the form of detection of sitting and standing with 100% accuracy. The results of the face and the eyes detection to classificate focus or not focus have 98.33% accuracy.

Keywords: *Focus Detection, Camera supervisor, Haar Cascade Classifier, Raspberry Pi*

DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	v
<i>ABSTRACT</i>	vi
DAFTAR ISI.....	vii
DAFTAR TABEL	ix
DAFTAR GAMBAR.....	x
BAB 1 PENDAHULUAN	Error! Bookmark not defined.
1.1 Latar Belakang.....	Error! Bookmark not defined.
1.2 Rumusan Masalah.....	Error! Bookmark not defined.
1.3 Tujuan Penelitian	Error! Bookmark not defined.
1.4 Manfaat Penelitian.....	Error! Bookmark not defined.
1.5 Batasan Masalah.....	Error! Bookmark not defined.
1.6 Sistematika Pembahasan	Error! Bookmark not defined.
BAB 2 LANDASAN KEPUSTAKAAN	Error! Bookmark not defined.
2.1 Tinjauan Pustaka	Error! Bookmark not defined.
2.2 Dasar Teori	Error! Bookmark not defined.
2.2.1 Perhatian.....	Error! Bookmark not defined.
2.2.2 HSV (<i>Hue Saturation Value</i>)	Error! Bookmark not defined.
2.2.3 <i>Thresholding</i>	Error! Bookmark not defined.
2.2.4 <i>Haar Cascade Clasifier</i>	Error! Bookmark not defined.
2.2.5 Raspberry Pi	Error! Bookmark not defined.
2.2.6 Kamera Logitech C270	Error! Bookmark not defined.
2.2.7 Buzzer	Error! Bookmark not defined.
2.2.8 Sistem Operasi Raspbian.....	Error! Bookmark not defined.
2.2.9 OpenCv.....	Error! Bookmark not defined.
BAB 3 METODOLOGI PENELITIAN	Error! Bookmark not defined.
3.1 Studi Literatur	Error! Bookmark not defined.
3.2 Rekayasa Kebutuhan.....	Error! Bookmark not defined.
3.3 Perancangan Sistem.....	Error! Bookmark not defined.
3.4 Implementasi	Error! Bookmark not defined.
3.5 Pengujian dan analisis Sistem	Error! Bookmark not defined.
3.6 Pengambilan Kesimpulan.....	Error! Bookmark not defined.
BAB 4 REKAYASA KEBUTUHAN.....	Error! Bookmark not defined.

4.1	Deskripsi Umum	Error! Bookmark not defined.
4.1.1	Perspektif Sistem	Error! Bookmark not defined.
4.1.2	Ruang Lingkup	Error! Bookmark not defined.
4.1.3	Karakteristik Pengguna	Error! Bookmark not defined.
4.1.4	Lingkungan Operasi Sistem	Error! Bookmark not defined.
4.1.5	Batasan Perancangan dan Implementasi.....	Error! Bookmark not defined.
4.1.6	Asumsi dan Ketergantungan	Error! Bookmark not defined.
4.2	Rekayasa Kebutuhan.....	Error! Bookmark not defined.
4.2.1	Kebutuhan Fungsional	Error! Bookmark not defined.
4.2.2	Kebutuhan Perangkat Keras.....	Error! Bookmark not defined.
4.2.3	Kebutuhan Perangkat Lunak	Error! Bookmark not defined.
BAB 5	PERANCANGAN DAN IMPLEMENTASI	Error! Bookmark not defined.
5.1	Perancangan Sistem.....	Error! Bookmark not defined.
5.1.1	Perancangan Perangkat Keras	Error! Bookmark not defined.
5.1.2	Perancangan Perangkat Lunak.....	Error! Bookmark not defined.
5.2	Implementasi Sistem.....	Error! Bookmark not defined.
5.2.1	Implementasi Perangkat Keras	Error! Bookmark not defined.
5.2.2	Implementasi Perangkat Lunak.....	Error! Bookmark not defined.
BAB 6	PENGUJIAN DAN ANALISIS SISTEM.....	Error! Bookmark not defined.
6.1	Pengujian dan Analisis Akurasi Deteksi Posisi Duduk dan Berdiri Operator Menggunakan HSV Dan Tresholding.....	Error! Bookmark not defined.
6.1.1	Tujuan	Error! Bookmark not defined.
6.1.2	Alat	Error! Bookmark not defined.
6.1.3	Metode.....	Error! Bookmark not defined.
6.1.4	Hasil.....	Error! Bookmark not defined.
6.1.5	Analisis	Error! Bookmark not defined.
6.2	Pengujian dan Analisis Akurasi Deteksi Wajah dan Deteksi Mata Manusia Menggunakan Haar-Cascade Classifier.....	Error! Bookmark not defined.
6.2.1	Tujuan	Error! Bookmark not defined.
6.2.2	Alat	Error! Bookmark not defined.
6.2.3	Metode.....	Error! Bookmark not defined.
6.2.4	Hasil.....	Error! Bookmark not defined.
6.2.5	Analisis	Error! Bookmark not defined.
BAB 7	PENUTUP	Error! Bookmark not defined.
7.1	Kesimpulan.....	Error! Bookmark not defined.
7.2	Saran	Error! Bookmark not defined.
	Daftar Pustaka.....	Error! Bookmark not defined.

DAFTAR TABEL

Tabel 2.1 Perbedaan penelitian sebelumnya dengan penelitian saat ini.....	Error! Bookmark not defined.
Tabel 2.2 Spesifikasi Raspberry Pi 3 Model B.....	Error! Bookmark not defined.
Tabel 2.3 Spesifikasi Kamera Logitech C270	Error! Bookmark not defined.
Tabel 2.4 Spesifikasi Buzzer	Error! Bookmark not defined.
Tabel 4.1 Kebutuhan fungsional perangkat keras	Error! Bookmark not defined.
Tabel 4.2 Kebutuhan perangkat lunak	Error! Bookmark not defined.
Tabel 5.1 Koneksi pin perancangan perangkat keras	Error! Bookmark not defined.
Tabel 5.2 Ketentuan kondisi subjek memperhatikan monitor	Error! Bookmark not defined.
Tabel 5.3 Perancangan keluaran sistem dengan buzzer	Error! Bookmark not defined.
Tabel 5.4 Include file yang dibutuhkan sistem deteksi perhatian operator	Error! Bookmark not defined.
Tabel 5.5 Kode Program inialisasi variabel	Error! Bookmark not defined.
Tabel 5.6 Kode program konversi Thresholding	Error! Bookmark not defined.
Tabel 5.7 kode program deteksi duduk dan berdiri..	Error! Bookmark not defined.
Tabel 5.8 Kode untuk memanggil library deteksi wajah dan mata	Error! Bookmark not defined.
Tabel 5.9 Kode program untuk deteksi wajah dan mata	Error! Bookmark not defined.
Tabel 5.10 Kode program untuk klasifikasi perhatian	Error! Bookmark not defined.
Tabel 5.11 Kode program untuk mengaktifkan fungsi buzzer	Error! Bookmark not defined.
Tabel 6.1 Hasil pengujian deteksi duduk dan berdiri	Error! Bookmark not defined.
Tabel 6.2 Hasil pengujian deteksi duduk dan berdiri subjek ..	Error! Bookmark not defined.
Tabel 6.3 Gambar hasil deteksi wajah dan mata	Error! Bookmark not defined.
Tabel 6.4 Hasil Perancangan klasifikasi perhatian subjek terhadap monitor..	Error! Bookmark not defined.
Tabel 6.5 Kesesuaian hasil klasifikasi sistem saat mendeteksi perhatian subjek terhadap monitor.....	Error! Bookmark not defined.
Tabel 6.6 Gambar hasil deteksi wajah dan mata ketika kepala miring	Error! Bookmark not defined.
Tabel 6.7 Hasil deteksi wajah dan mata ketika kepala miring	Error! Bookmark not defined.
Tabel 6.8 Gambar hasil akurasi jarak deteksi wajah dan deteksi mata	Error! Bookmark not defined.
Tabel 6.9 Hasil akurasi jarak deteksi wajah dan deteksi mata	Error! Bookmark not defined.



DAFTAR GAMBAR

Gambar 2.1 Haar-like feature.	Error! Bookmark not defined.
Gambar 2.2 Proses cascade classifier	Error! Bookmark not defined.
Gambar 2.3 Raspberry Pi 3 Model B	Error! Bookmark not defined.
Gambar 2.4 Skema BCM2837	Error! Bookmark not defined.
Gambar 2.5 Kamera Logitech C270.....	Error! Bookmark not defined.
Gambar 2.6 Buzzer	Error! Bookmark not defined.
Gambar 2.7 Sistem Operasi Raspbian	Error! Bookmark not defined.
Gambar 2.8 Logo Software OpenCV	Error! Bookmark not defined.
Gambar 3.1 Diagram blok alur metodologi penelitian	Error! Bookmark not defined.
Gambar 3.2 Bagan rekayasa kebutuhan	Error! Bookmark not defined.
Gambar 3.3 Blok diagram desain sistem.....	Error! Bookmark not defined.
Gambar 5.1 Skema perancangan perangkat keras ...	Error! Bookmark not defined.
Gambar 5.2 Diagram alir proses sistem secara keseluruhan..	Error! Bookmark not defined.
Gambar 5.3 Cara kerja deteksi duduk atau berdiri dengan Thresholding	Error! Bookmark not defined.
Gambar 5.4 Proses deteksi wajah menggunakan <i>Haar cascade classifier</i>	Error! Bookmark not defined.
Gambar 5.5 Fitur kotak untuk mendeteksi wajah dan mata ..	Error! Bookmark not defined.
Gambar 5.6 Penerapan perhitungan pada fitur kotak.....	Error! Bookmark not defined.
Gambar 5.7 Contoh perhitungan integral image	Error! Bookmark not defined.
Gambar 5.8 Proses yang dilalui gambar untuk mendeteksi wajah	Error! Bookmark not defined.
Gambar 5.9 Implementasi perangkat keras sistem ..	Error! Bookmark not defined.

BAB 1 PENDAHULUAN

1.1 Latar Belakang

Dalam konteks interaksi manusia, memahami perilaku manusia merupakan salah satu masalah yang menantang pada saat ini. Pengenalan perilaku manusia telah menjadi topik yang penting dalam mendeteksi kegiatan emosional dengan banyak aplikasi dasar, seperti pada bidang robotika, pengawasan video, dan interaksi antar manusia dan komputer (Shirbhate & Talele, 2016). Perilaku manusia sangat menarik untuk dipelajari karena manusia berperilaku sesuai emosi mereka. Kegiatan emosional yaitu berupa respon ungkapan dari emosi manusia, salah satunya adalah perilaku tidak memperhatikan. Kegiatan emosional manusia terutama tingkat perhatian manusia dapat dikembangkan menjadi dasar sebuah teknologi pengawasan video.

Pada zaman ini, telah banyak yang menerapkan sistem keamanan menggunakan kamera pengawas dengan operator sebagai pengawas monitor. Operator diwajibkan melakukan pengawasan monitor di dalam ruangan dengan durasi shift rata-rata 6 jam sampai 8 jam. Sebuah riset yang melibatkan Dr. Craig Donald dengan rekannya untuk mencari tipe faktor operator kamera pengawas, dimana para peserta dalam penelitian diharuskan melakukan tugas-tugas deteksi video berdurasi 90 menit dengan berbagai insiden dan berbagai tingkat kesulitan. Para peserta tidak diberikan istirahat selama periode ini. Pengujian efektifitas saat melihat monitor, bahkan disaat kondisi sulit seperti harus melihat kamera pengawas terus-menerus hingga 90 menit. Dr. Craig menemukan bahwa seseorang akan melakukan pekerjaannya dengan baik di 30 menit pertama pada percobaan selama 90 menit, sedangkan setengah dari responden hanya bertahan selama 15 menit (Donald, 2015). Dengan durasi kerja operator yang terbilang cukup lama, banyak operator yang kurang perhatian sehingga tugas pengawasannya menjadi kurang maksimal. Karena tidak adanya pengawas kinerja operator untuk mengetahui tingkat perhatian operator, maka teknologi pengawas operator secara realtime dibutuhkan.

Dewasa ini, telah banyak teknologi real-time yang menggunakan perilaku manusia sebagai sebuah *input* dari sistem. Sebuah penelitian yang dilatarbelakangi ketika operator dihadapkan dengan stress, pandangan mereka, cara berpikir dan pengambilan keputusan mulai melemah, serta meningkatkan kesempatan kesalahpahaman peristiwa yang terjadi dan menambah potensi kesalahan yang disebabkan oleh operator. Algoritma yang digunakan adalah *Bidirectional Long Short Term Memory network* (BiD-LSTM) dan *Deep Convolutional Neural Network* (DCNN) untuk memproses *audio* dan data wajah, maka peneliti dapat menghasilkan pengenalan pola emosi dari operator (Rybak, et al., 2017).

Berdasarkan masalah yang telah disebutkan pada paragraf sebelumnya dan terinspirasi dari penelitian yang ada, maka penulis melakukan penelitian mengenai “Sistem Deteksi Perhatian Operator Kamera Pengawas Terhadap monitor Menggunakan Haar-cascade classifier”. Sistem ini menggunakan kamera sebagai penangkap citra dan Raspberry Pi sebagai pemroses input dari kamera, output dari sistem berupa buzzer dimana jika operator tidak memperhatikan monitor akan berbunyi. Diharapkan sistem ini dapat membantu mengawasi kinerja operator agar lebih efisien.

Untuk mengetahui kondisi memperhatikan monitor operator pengawas kamera, sistem perlu mengetahui posisi dan kondisi operator pengawas kamera. Posisi yang dimaksud adalah posisi duduk atau berdiri operator pengawas kamera, sedangkan kondisi yang dimaksud pada penelitian ini adalah kondisi kedua mata dan posisi wajah operator pengawas kamera. Dalam mengenali posisi duduk dan berdiri, citra digital RGB diubah ke HSV untuk mengambil rentang warna kulit manusia kemudian dilakukan threshold dan perhitungan piksel putih untuk menentukan posisi duduk atau berdiri. Jika operator ditentukan dalam posisi duduk, proses akan dilanjutkan ke tahap deteksi wajah dan mata. Untuk deteksi wajah dan mata operator pengawas kamera, digunakan algoritma Haar cascade classifier.

Algoritma yang akan digunakan untuk deteksi wajah dan mata adalah *Haar Cascade Classifier* karena metode ini efektif untuk deteksi objek, terutama wajah manusia yang memiliki ciri berbeda-beda. *Haar Cascade Classifier* dapat digunakan secara luas selain mengenali wajah, juga memiliki komputasi sederhana dan efektif sehingga tidak memberatkan kinerja sistem. Metode ini dapat digunakan untuk mendeteksi wajah dan sepasang mata operator kamera pengawas. Menurut Hadi Santoso dan Agus Harjoko (2013), algoritma ini dapat mendeteksi wajah selain menghadap lurus ke depan juga dapat mendeteksi wajah hingga rotasi sebesar 15° (kekanan, kekiri, keatas, dan menunduk) (Santoso & Harjoko, 2013).

1.2 Rumusan Masalah

Berdasarkan latar belakang penelitian yang telah dijelaskan sebelumnya, maka rumusan masalah adalah sebagai berikut:

1. Bagaimana membangun sistem deteksi perhatian operator kamera pengawas terhadap monitor
2. Bagaimana implementasi pengolahan citra menggunakan HSV dan thresholding untuk deteksi posisi duduk dan berdiri
3. Bagaimana implementasi algoritma Haar-cascade classifier untuk deteksi wajah dan deteksi mata
4. Bagaimana akurasi sistem dalam mendeteksi perhatian operator kamera pengawas terhadap monitor

1.3 Tujuan Penelitian

Berdasarkan latar belakang penelitian yang telah dijelaskan sebelumnya, maka tujuan dari penelitian ini adalah sebagai berikut:

1. Membangun sistem deteksi perhatian operator kamera pengawas terhadap monitor
2. Mengimplementasikan pengolahan citra menggunakan HSV dan thresholding untuk mengenali posisi duduk dan berdiri
3. Mengimplementasikan algoritma Haar-cascade classifier untuk mengenali wajah dan mata
4. Mengetahui akurasi sistem dalam mendeteksi perhatian operator kamera pengawas terhadap monitor

1.4 Manfaat Penelitian

Berdasarkan latar belakang penelitian yang telah dijelaskan sebelumnya, maka manfaat dari diadakannya penelitian ini adalah sebagai berikut:

1. Tingkat perhatian setiap manusia berbeda, dengan adanya sistem ini operator dapat mengetahui rata-rata tingkat perhatian maksimalnya untuk kemudian dilakukan pengaturan jam kerja
2. Pemimpin operator dapat memantau keadaan operator tanpa harus mengawasi secara intens
3. Masyarakat dapat menggunakan alat ini untuk mengawasi tingkat perhatian manusia dalam ruangan selain operator kamera pengawas
4. Bagi pihak kampus, penelitian ini dapat dikembangkan untuk keperluan yang lebih universal
5. Bagi peneliti, penelitian ini dapat memenuhi sebagian persyaratan kurikulum Fakultas Ilmu Komputer

1.5 Batasan Masalah

Berikut ini adalah batasan masalah pada penelitian sistem deteksi tingkat perhatian operator pengawas kamera:

1. Sistem diujikan dalam ruangan tertutup
2. Sistem diujikan dengan pencahayaan ruangan 200 sampai 350 LUX
3. Sistem dapat mendeteksi posisi subjek maksimal satu orang
4. Jarak optimal sistem dalam mendeteksi wajah subjek adalah 40cm sampai 100cm

1.6 Sistematika Pembahasan

Sistematika pembahasan digunakan untuk membahas penelitian ini secara garis besar. Berikut ini adalah uraian dari sistematika pembahasan:

Bab I Pendahuluan

Bab ini membahas mengenai latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, dan sistematika pembahasan dari penelitian sistem Deteksi Perhatian Operator Kamera Pengawas Terhadap monitor.

Bab II Landasan Kepustakaan

Pada bab ini akan dijelaskan mengenai studi literatur, sumber-sumber terkait dan dasar teori yang akan digunakan untuk menunjang berjalannya penelitian sistem deteksi tingkat perhatian operator kamera pengawas.

Bab III Metodologi

Pada Bab Metodologi akan dijelaskan metode dan langkah-langkah yang akan digunakan untuk keperluan pengerjaan sistem Deteksi Perhatian Operator Kamera Pengawas Terhadap monitor. Metode yang digunakan yaitu studi literatur mengenai penelitian yang terkait dan kebutuhan untuk penelitian, analisis kebutuhan, desain sistem dan implementasi, pengujian dan analisis sistem, dan pengambilan kesimpulan.

Bab IV Rekayasa Kebutuhan

Bab ini memuat kebutuhan yang harus dipenuhi guna keberhasilan berjalannya sistem sesuai dengan tujuan dibuatnya sistem. Rincian dari bab ini berupa deskripsi secara umum dan rekayasa kebutuhan.

Bab V Perancangan dan Implementasi

Bab Perancangan dan implementasi membahas perancangan sistem baik secara hardware dan software sebelum kemudian diimplementasikan menjadi sistem Deteksi Perhatian Operator Kamera Pengawas Terhadap monitor. Implementasi pada bab ini berupa implementasi pengolahan citra menggunakan HSV dan *Thresholding* dan implementasi algoritma yang digunakan yaitu Haar-cascade classifier. Selain implementasi algoritma, akan dilakukan tahap klasifikasi tingkat perhatian untuk mendapatkan keluaran.

Bab VI Pengujian dan Analisis Sistem

Bab pengujian dan analisis berisi hasil dari pengujian sistem dan analisis sistem. Pengujian sistem dilakukan baik secara bertahap maupun secara keseluruhan sistem. Analisis yang dilakukan terhadap sistem dilakukan setelah tahap pengujian. Pengujian dan analisis dilakukan pada hardware, deteksi posisi (duduk dan berdiri), deteksi wajah, dan sistem secara keseluruhan.

Bab VII Penutup

Bab penutup membahas mengenai kesimpulan dan saran yang dihasilkan dari keseluruhan proses pengerjaan penelitian sistem deteksi perhatian operator engawas kamera yang telah dilakukan.

BAB 2 LANDASAN KEPUSTAKAAN

2.1 Tinjauan Pustaka

Dalam pelaksanaan penelitian, peneliti perlu mencari informasi mengenai penelitian relevan yang telah ada sebelumnya sebagai bahan acuan pengerjaan penelitian. Pada Tabel 2.1 membahas penelitian sebelumnya yang berkaitan dengan pengerjaan penelitian “Sistem Deteksi Perhatian Operator Kamera Pengawas Terhadap Monitor Menggunakan Haar-cascade classifier”.

Tabel 2.1 Perbedaan penelitian sebelumnya dengan penelitian saat ini

No	Judul	Algoritma dan Metode	Arah Penelitian	Pembandingan
1	Detection and Controlling of Drivers' Visual Focus of Attention (Debnath, et al., 2017)	Teknik <i>Gaze fixation</i>	Deteksi perhatian pengemudi menggunakan tatapan mata dan arah gerak kepala	Objek deteksi berupa tatapan mata dan arah gerak kepala
2	The design of intelligent crowd attention detection system based on face detection technology (Qiming, et al., 2017)	<i>Haar-like features</i> dan algoritma <i>Adaboost</i>	Deteksi arah perhatian sekumpulan orang	Objek deteksi berupa wajah sekumpulan orang
3	Evaluation of head pose features for stress detection and classification (Giannakakis, et al., 2018)	<i>Head movements estimation</i> dan <i>Head pose estimation in 3D model</i>	Menyelidiki fitur pose kepala dalam respon stress tertentu	Perhatian utama berupa pose kepala dan sudut rotasi kepala

Penjelasan dari Tabel 2.1, pada penelitian pertama inti penelitian ini berada pada deteksi perhatian pengemudi menggunakan arah pandangan mata dan pergerakan kepala (Debnath, et al., 2017). Dari penelitian ini, penelitian penulis juga mengenai deteksi perhatian manusia dimana gerakan kepala manusia menjadi salah satu objek penelitian. Perbedaan penelitian penulis dan penelitian ini yaitu subjek penelitian Pratim adalah pengemudi dan objek pendukung deteksi perhatiannya berupa arah pandangan mata menggunakan pupil mata. Penelitian Debnath, et al. (2017) menggunakan *gaze detection* dimana pemrosesan yang dilakukan sistem cukup berat, sehingga perlu menggunakan laptop yang berukuran cukup besar. Sedangkan penelitian sistem deteksi perhatian operator cukup menggunakan perangkat Raspberry.

Penelitian kedua yang dilakukan oleh Qiming, et al. (2017), Penelitian ini berpusat pada deteksi wajah untuk mengetahui dan menganalisa pusat perhatian dari sekelompok orang. Persamaan penelitian ini dengan penelitian penulis yaitu tentang pengenalan wajah menggunakan metode Haar like features dan algoritma Adaboost untuk mengetahui arah perhatian atau perhatian manusia (Qiming, et al., 2017). Perbedaan penelitian penulis dan penelitian ini terletak pada subjek penelitian, dimana penelitian Qiming dapat mendeteksi sekumpulan orang sedangkan penelitian penulis hanya mendeteksi satu subjek. Sistem ini menggunakan komputer sebagai pemroses sistem, sedangkan sistem deteksi perhatian operator menggunakan Raspberry Pi yang terpisah dari komputer sehingga tidak perlu mengganggu kinerja komputer pengawas.

Penelitian keempat yang dilakukan oleh Giannakakis, et al., (2018), berpusat pada Deteksi stress manusia menggunakan fitur pose kepala dan kecepatan gerak kepala. Dari penelitian Giannakakis dan rekannya, penulis juga menggunakan palangan kepala untuk deteksi perhatian operator kamera pengawas terhadap monitor (Giannakakis, et al., 2018). Perbedaan penelitian penulis dan penelitian ini yaitu, penelitian ini menekankan pada pose kepala terutama pada gerak *translational* dan *rotational* dari kepala manusia untuk mendeteksi stress. Penelitian ini dapat digunakan untuk mendeteksi stress, sedangkan penelitian deteksi perhatian tidak dapat digunakan untuk mendeteksi stress.

2.2 Dasar Teori

Dasar teori akan menjelaskan mengenai dasar-dasar teori, perhitungan, dan keterangan mengenai kebutuhan hardware dan software. Teori yang dijelaskan berasal dari beberapa sumber pustaka seperti buku, jurnal, internet dengan sumber yang valid, dan penelitian sebelumnya yang terkait.

2.2.1 Perhatian

Perhatian dapat didefinisikan dalam berbagai bidang. Dalam Kamus Besar Bahasa Indonesia, kata perhatian diartikan sebagai minat pada suatu objek (Setiawan, 2012). Jadi, perhatian dapat diartikan sebagai menunjukan pandangan pada suatu objek. Perhatian seseorang pada suatu objek dapat digunakan untuk menilai kinerja seseorang.

Perhatian yaitu memusatkan kesadaran pada rangsangan yang ada (rangsangan terbatas). Perhatian merupakan suatu proses aktif, dimana jika seseorang melonggarkan perhatiannya, maka seseorang tersebut dapat terseret pada pikiran-pikiran yang tidak relevan. Terlebih jika seseorang tersebut dalam situasi dimana stimulus lain bersaing untuk mendapat perhatian (Semiun, 2006).

Perhatian yang menjadi objek penelitian pada penelitian deteksi perhatian operator pengawas kamera yaitu minat atau pandangan yang dipusatkan pada suatu objek. Dimana minat atau pandangan subjek yang dipusatkan, dan objek pusatnya adalah monitor.

2.2.2 HSV (*Hue Saturation Value*)

Deteksi kulit manusia dimulai dari pengenalan area dan piksel warna kulit pada setiap frame. Warna kulit manusia digunakan sebagai dasar untuk mendeteksi kulit manusia. Pengenalan warna kulit manusia pada penelitian ini menggunakan ruang warna HSV, karena HSV lebih intuitif daripada RGB. Setiap frame dari video mengandung format warna RGB. RGB termasuk model warna aditif yang berfungsi sebagai penginderaan gambar dalam tampilan visual pada peralatan elektronik seperti komputer dan fotografi. Format warna ini digunakan sebagai tampilan pada layar monitor komputer karena warna latar belakang layar monitor adalah hitam (Kolkur, et al., 2016). Penggunaan ruang warna HSV pada penelitian deteksi perhatian operator kamera pengawas terhadap monitor digunakan untuk menyeleksi piksel yang mengandung warna kulit. Piksel dengan warna kulit digunakan sebagai batas threshold untuk operasi *Thresholding* berikutnya.

HSV merupakan salah satu ruang warna yang mengandung unsur warna hue yang dilambangkan dengan (H), saturasi yang dilambangkan dengan (S), dan Intensitas yang dilambangkan dengan (V). Unsur warna Hue yang dimaksud pada ruang warna HSV menggambarkan perubahan warna dari merah ke hijau, dan saturasi menggambarkan perubahan warna dari merah ke merah muda, sedangkan intensitas atau *value* mewakili perubahan intensitas pencahayaan dari hitam ke putih. Jika dibandingkan dengan ruang warna RGB, HSV dapat menghasilkan pengolahan citra digital lebih baik untuk mendeteksi kulit manusia (Rahman, et al., 2014).

Transformasi dari ruang warna RGB ke ruang warna HSV dapat didefinisikan pada persamaan berikut:

$$H' = \begin{cases} 0 & \text{jika } C = 0 \\ \frac{G-B}{C} \bmod 6 & \text{jika } Max = R \\ \frac{B-R}{C} + 2 & \text{jika } Max = G \\ \frac{R-G}{C} + 4 & \text{jika } Max = B \end{cases} \quad (2.1)$$

$$H = 60^\circ \times H' \quad (2.2)$$

$$S = \begin{cases} 0 & \text{jika } V = 0 \\ \frac{C}{V} & \text{jika } v \neq 0 \end{cases} \quad (2.3)$$

$$V = \text{Max}(R, G, B) \quad (2.4)$$

Dimana berdasarkan persamaan (2.1) dan persamaan (2.4), Max adalah nilai maksimal dari piksel R, G atau B. Sedangkan C berasal dari nilai Max yang dikurangi nilai minimum dari piksel R, G atau B. Hasil dari persamaan (2.1), yaitu nilai H' digunakan pada persamaan (2.2) untuk menemukan nilai hue. Persamaan (2.3) digunakan untuk menemukan nilai saturasi dan persamaan (2.4) digunakan untuk menemukan nilai value (Zhang, et al., 2018).

2.2.3 Thresholding

Ide dasar dari *thresholding* yaitu untuk memilih nilai ambang tingkat keabu-abuan yang optimal untuk memisahkan objek utama pada gambar dari latar belakang berdasarkan distribusi tingkat keabu-abuannya. *Thresholding* menciptakan gambar biner dari tingkat keabu-abuan dengan mengubah semua piksel dibawah ambang ke biner nol dan piksel lainnya ke biner satu. *Thresholding* citra HSV pada penelitian ini digunakan untuk mempermudah dalam menentukan subjek dalam posisi duduk atau berdiri.

Jika $g(x, y)$ adalah versi ambang dari $f(x, y)$ di beberapa batas global T , maka nilai $g(x, y)$ didefinisikan pada Persamaan (2.5) sebagai:

$$\begin{aligned} \text{Nilai } g(x, y) &= 1 \text{ jika } f(x, y) \geq 0 \\ &= 0 \text{ (jika tidak)} \end{aligned} \quad (2.5)$$

Operasi *thresholding* didefinisikan pada Persamaan (2.6) berikut.

$$T = M [x, y, p(x, y), f(x, y)] \quad (2.6)$$

Dimana, T adalah ambang batas, $f(x, y)$ merupakan nilai abu-abu dari titik (x, y) , dan $p(x, y)$ menunjukkan *local property* dari rata-rata nilai keabu-abuan yang berpusat pada titik (x, y) (Vala & Baxi, 2013).

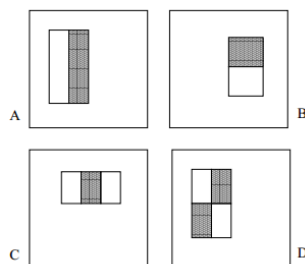
Thresholding pada penelitian ini digunakan untuk binerisasi citra HSV. Citra hsv dengan piksel yang termasuk dalam rentang warna kulit akan diubah menjadi piksel putih. Binerisasi ini dilakukan untuk menentukan posisi duduk dan berdiri. Cara yang digunakan untuk menentukan posisi yaitu dengan membagi frame menjadi dua bagian dan menghitung piksel warna putih.

2.2.4 Haar Cascade Clasifier

Algoritma yang digunakan untuk deteksi wajah dan mata pada penelitian ini adalah Algoritma *Haar Cascade Clasifier* karena Algoritma ini efektif untuk deteksi objek, terutama wajah manusia yang memiliki ciri berbeda-beda. Menurut Santoso dan Harjoko, Algoritma ini dapat mendeteksi wajah selain menghadap lurus ke depan kamera juga dapat mendeteksi wajah hingga rotasi sebesar 15° (kekanan, kekiri, keatas, dan menunduk). Algoritma ini diusulkan pada jurnal Paul Viola dan Michael Jones yang berjudul "*Rapid Object Detection using a Boosted Cascade of Simple Features*" pada tahun 2001. Algoritma ini merupakan sebuah machine learning yang berbasis pada pendekatan fungsi cascade yang dilatih dari banyak gambar positif dan negatif (Santoso & Harjoko, 2013).

Proses menemukan wajah dimulai dari menentukan *haar feature* untuk mendeteksi objek. Proses menentukan haar feature menggunakan *haar-like features* seperti pada Gambar 2.1, dimana kita harus melakukan pembelajaran data objek yang dideteksi agar menghasilkan sebuah pohon keputusan dengan nama *cascade classifier*. Pohon keputusan *Cascade classifier* digunakan sebagai penentu apakah ada atau tidak objek tertentu dalam setiap frame yang diproses. *Haar-like feature* menggolongkan gambar kedalam kotak-kotak yang didalamnya terdapat piksel dengan nilai- nilai tertentu. Setiap kotak akan di proses untuk

mendapat nilai *threshold*. Nilai *threshold* digunakan untuk menentukan parameter klasifikasi objek, apakah objek dideteksi sebagai wajah atau bukan wajah (Santoso & Harjoko, 2013).



Gambar 2.1 Haar-like feature.

Sumber: (Viola & Jones, 2001)

Haar-like ditentukan dari pengurangan pixel pada rata-rata daerah gelap dari pixel pada rata-rata daerah terang. Jika perbedaan ada diatas *threshold* (diatur selama masa pembelajaran), maka fitur tersebut dapat dikatakan ada. Untuk menentukan adanya *Haar feature* di setiap lokasi gambar, Viola dan Jones menggunakan teknik *intermediate representation* untuk gambar yang dipanggil sebagai *integral image*. *Integral image* digunakan untuk melakukan perhitungan dan penjumlahan piksel pada video. Gambar integral pada lokalisasi x, y , berisi jumlah dari piksel diatas dan ke kiri dari x, y . Persamaan (2.7) berikut ini merupakan persamaan integral image.

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y') \quad (2.7)$$

Berdasarkan persamaan (2.9), dimana gambar pada lokasi x, y berisi jumlah dari piksel dibawah, dan dikiri x, y . $ii(x, y)$ adalah gambar integral dan $i(x, y)$ adalah gambar asli. Menggunakan operasi berikut:

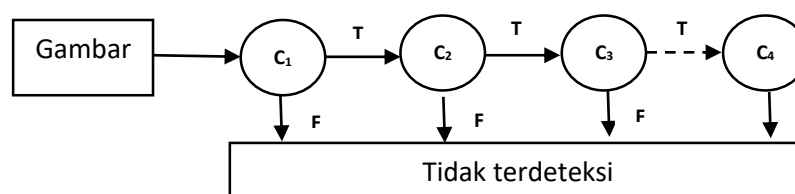
$$s(x, y) = s(x, y - 1) + i(x, y) \quad (2.8)$$

$$ii(x, y) = ii(x - 1, y) + s(x, y) \quad (2.9)$$

Pada Persamaan (2.8), $s(x, y)$ adalah jumlah baris kumulatif $s(x, -1) = 0$ dan $ii(x - 1, y) = 0$) gambar integral bisa dihitung sekali operasi dari gambar asli. Nilai integral pada setiap pixel adalah penjumlahan dari semua pixel di atasnya dan di kirinya (dimulai dari kiri atas sampai ke kanan bawah), operasi ini dapat disebut juga sebagai operasi matematika per-pixel (Viola & Jones, 2001).

Setelah menghitung integral image yaitu memilih fitur Haar yang digunakan dan juga mengubah nilai *threshold* menggunakan metode *machine-learning* bernama *AdaBoost*. Metode ini digunakan untuk mendapatkan *cascade classifier*. Metode *AdaBoost* menggabungkan banyak classifier untuk dapat membuat satu classifier. Pada saat setiap *Adaboost classifier* menetapkan suatu nilai, maka gabungan dari nilai-nilai inilah yang akan membentuk suatu classifier yang lebih baik. Dengan menggabungkan rangkaian *AdaBoost classifier* sebagai rantai filter, maka setiap filter merupakan *AdaBoost classifier* terpisah dengan *weak classifier*

berjumlah sedikit dan sama. Filter yang terdapat pada setiap level akan dilatih mengklasifikasikan citra yang sebelumnya telah disaring dengan *training set* (database dari wajah). Jika salah satu dari proses filter gagal, maka daerah pada gambar akan digolongkan sebagai bukan wajah. Dan jika filter berhasil melewati daerah gambar, maka gambar akan masuk pada filter selanjutnya. Daerah gambar yang dapat melalui seluruh proses filter akan digolongkan sebagai wajah (Krishna & A. Srinivasulu, 2012).



Gambar 2.2 Proses cascade classifier

Proses *cascade classifier* adalah proses klasifikasi bertumpuk untuk meningkatkan efisiensi proses klasifikasi. Berdasarkan Gambar 2.2 klasifikasi pada tingkat pertama dimana setiap subgambar digolongkan menggunakan satu fitur, lalu menghasilkan nilai T (*true*) jika gambar memenuhi fitur Haar tertentu dan F (*false*) jika tidak memenuhi fitur. Klasifikasi pertama akan mengurangi beban klasifikasi pada tahap dua sebanyak kurang lebih 50% subgambar untuk diklasifikasikan. Hasil dari klasifikasi tahap dua berupa nilai T (*True*) jika gambar memenuhi proses integral image dan F (*false*) jika tidak memenuhi. Dengan bertambahnya tingkat klasifikasi, maka diperlukan syarat yang lebih rinci dari sebelumnya sehingga fitur yang akan digunakan menjadi lebih banyak dan bervariasi. Jumlah subgambar yang lulus klasifikasi akan semakin berkurang hingga menggapai jumlah sekitar 2%. Hasil klasifikasi tingkat akhir berupa T (*True*) jika gambar memenuhi proses AdaBoost dan F (*false*) jika tidak memenuhi proses (Viola & Jones, 2001).

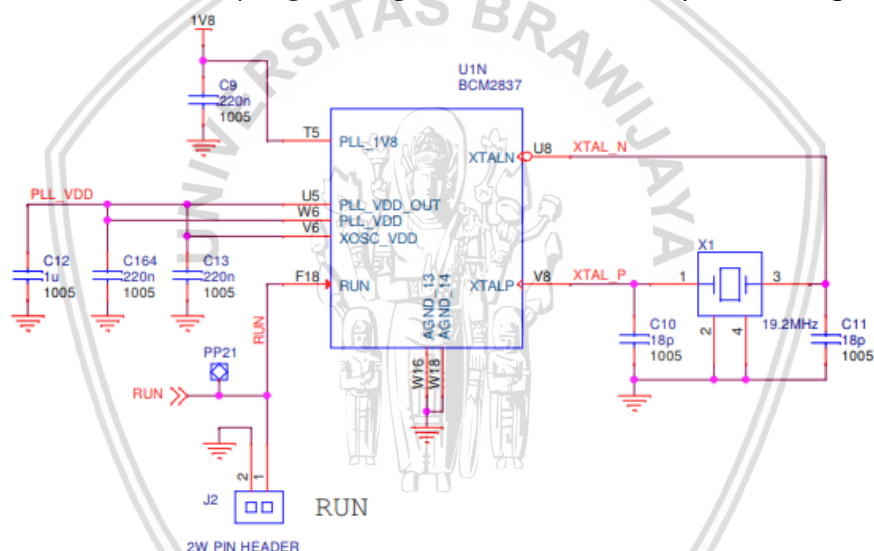
2.2.5 Raspberry Pi

Raspberry Pi 3 model B seperti pada Gambar 2.3 digunakan sebagai penerima data gambar dari kamera untuk kemudian dilakukan proses mengenali posisi (duduk atau berdiri) dan deteksi wajah. Hasil dari mengenali posisi duduk, berdiri dan deteksi wajah akan diklasifikasikan menjadi kondisi memperhatikan atau tidak. Kondisi memperhatikan monitor operator yang telah ditentukan akan memicu buzzer untuk berbunyi atau diam sebagai hasil keluaran sistem.



Gambar 2.3 Raspberry Pi 3 Model B
Sumber: (Raspberry Pi, 2016)

Raspberry Pi 3 model B memiliki bentuk fisik seperti pada Gambar 2.3. Pada Gambar 2.4 adalah skema dari prosesor raspberry pi 3 model B, yaitu BCM2837. Sistem ini menggunakan Raspberry karena mudah dalam maintenance dan mudah dipelajari. Sehingga pengguna dapat melakukan maintenance secara mandiri dan melakukan pengembangan sesuai kebutuhan perkembangan zaman.



Gambar 2.4 Skema BCM2837
Sumber: (RaspberryPi, 2016)

Spesifikasi dari Raspberry Pi 3 model B akan dijelaskan pada Tabel 2.2. Model 3B merupakan generasi ke-3 dari Raspberry Pi dan merupakan pengganti Raspberry Pi 2 model B.

Tabel 2.2 Spesifikasi Raspberry Pi 3 Model B

No	Spesifikasi	Keterangan
1	Processor	Quad Core 1.2GHz Broadcom BCM2837
2	CPU	64bit
3	RAM	1GB
4	wireless LAN	BCM43438
5	Bluetooth	Bluetooth Low Energy (BLE)
6	Pin extended GPIO	40 buah

No	Spesifikasi	Keterangan
7	Port USB 2	4 buah
8	Port Tambahan	4 Pole stereo output, Port composite video dan Full size HDMI
9	Konektor kamera	Port kamera CSI
10	Konektor display	Port display DSI
11	Port Memori	Port micro SD
12	Micro USB Power source	2.5 A

Sumber: (RaspberryPi, 2016)

2.2.6 Kamera Logitech C270

Kamera Logitech C270 pada Gambar 2.5 adalah sebuah kamera produksi dari Logitech. Kamera ini memiliki konektor berupa USB 2.0 sehingga dapat digunakan pada raspberry pi.



Gambar 2.5 Kamera Logitech C270

Sumber: (Logitech, 2018)

Perangkat ini digunakan untuk mengambil gambar operator secara real-time. Gambar yang diambil oleh kamera ini akan digunakan sebagai *input* untuk diproses di Raspberry Pi. Spesifikasi dari kamera logitech c270 akan dijelaskan pada Tabel 2.3.

Tabel 2.3 Spesifikasi Kamera Logitech C270

No	Spesifikasi	Keterangan
1	Video Capture	Hingga 1280 x 720 pixels
2	Foto	Hingga 3.0 megapixels
3	Resolusi Gambar	2592 x 1944
4	Konektor	USB 2.0 certified
5	Perangkat pendukung	Mic dengan noise reduction
6	Lensa	Fluid Crystal

Sumber: (Logitech, 2018)

2.2.7 Buzzer

Komponen buzzer seperti pada Gambar 2.6 terdiri dari transistor yang berfungsi sebagai saklar dan penguat arus. Cara kerja secara umum rangkaian buzzer adalah ketika sinyal keluar dari mikrokontroler dengan logika high, maka mikrokontroler akan mengirimkan sinyal kepada buzzer untuk bekerja. Ketika buzzer bekerja, buzzer akan menciptakan suara sesuai dengan pengaturan dan instruksi pemrograman pada mikrokontroler.



Gambar 2.6 Buzzer

Sumber: (Farnell, 2017)

Pada penelitian sistem deteksi perhatian operator pengawas kamera, buzzer digunakan sebagai keluaran dari sistem. Buzzer akan berbunyi jika keadaan operator sedang tidak memperhatikan monitor terhadap monitor. Pemakaian buzzer lebih efektif sebagai keluaran karena manusia cenderung lebih merespon peringatan dalam bentuk suara daripada bentuk visual. Spesifikasi dari buzzer akan dijelaskan pada Tabel 2.4.

Tabel 2.4 Spesifikasi Buzzer

No	Spesifikasi	Keterangan
1	Tegangan	6V DC
2	Tegangan Operasional	4 to 8V DC
3	<i>Rated Current*</i>	$\leq 30\text{mA}$
4	Keluaran suara pada 10cm*	$\geq 85\text{dB}$
5	Frekuensi Resonansi	$2300 \pm 300\text{Hz}$
6	Bunyi	Berlanjut
7	Temperatur Operasional	-25°C to $+80^{\circ}\text{C}$
8	Temperatur Penyimpanan	-30°C to $+85^{\circ}\text{C}$

Sumber: (Farnell, 2017)

2.2.8 Sistem Operasi Raspbian

Raspbian dengan lambang seperti pada Gambar 2.4 adalah sistem operasi berbasis pada Debian yang dikhususkan untuk perangkat keras Raspberry Pi. Sistem operasi ini merupakan program perangkat dasar yang membuat Raspberry Pi dapat beroperasi.



RASPBIAN

Gambar 2.7 Sistem Operasi Raspbian

Sumber: (Bytemark, 2017)

Raspbian memiliki 35.000 *packages* software dengan format yang tertata untuk kemudahan instalasi pada Raspberry Pi (Bytemark, 2017). Sistem operasi raspbian akan diinstal pada perangkat Raspberry Pi sebagai sistem operasi sebelum memasang software lain. Penulisan kode program dan menjalankan pemrosesan sistem dilakukan pada operasi raspbian.

2.2.9 OpenCv

OpenCV (*Open Source Computer Vision Library*) dengan logo seperti pada Gambar 2.8 dirilis dibawah lisensi BSD. Penggunaan OpenCV bebas untuk kebutuhan akademik maupun kebutuhan komersial.



Gambar 2.8 Logo Software OpenCV

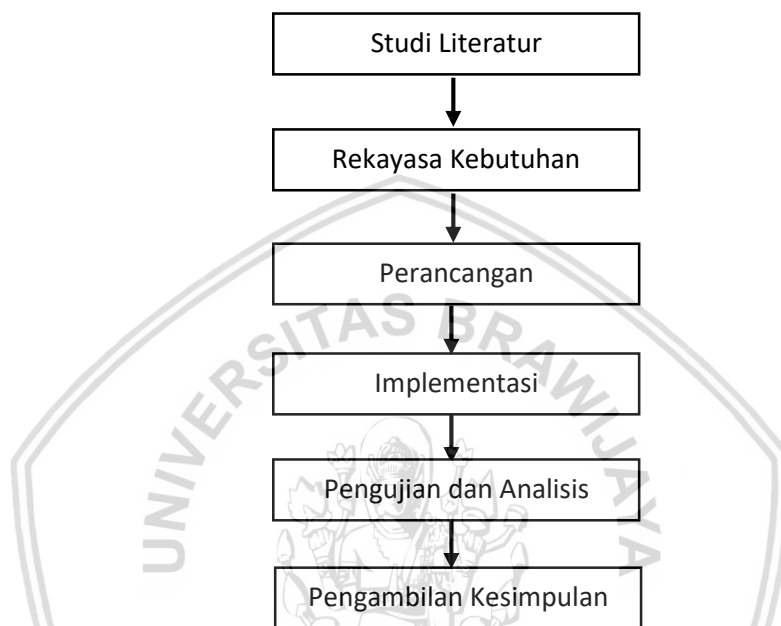
Sumber: (OpenCV, 2018)

Antar muka OpenCV dengan bahasa C, C++, Java dan Python yang mendukung Sistem Operasi Windows, Mac OS, Linux, Android dan iOS. OpenCV didesain untuk efisiensi komputasi dan perhatian yang kuat pada aplikasi real-time (OpenCV, 2018). Pada penelitian ini, OpenCV digunakan sebagai *library* yang di pasang pada Sistem Operasi Raspbian.



BAB 3 METODOLOGI PENELITIAN

Pada bab ini akan dijelaskan tahap-tahap yang akan dilakukan guna menyelesaikan penelitian. Pada Gambar 3.1 dijelaskan tahapan pengerjaan Sistem Deteksi Perhatian Operator Kamera Pengawas Terhadap Monitor Menggunakan Haar-cascade classifier dengan diagram blok.



Gambar 3.1 Diagram blok alur metodologi penelitian

Pada diagram blok Gambar 3.1 dijelaskan bahwa alur penelitian akan dimulai dari studi literatur untuk mengkaji ilmu-ilmu yang akan digunakan untuk penelitian. Tahap kedua yaitu rekayasa kebutuhan untuk mengetahui kebutuhan perancangan sistem, lalu akan dilanjutkan dengan perancangan sistem. Selanjutnya akan dilakukan implementasi, pengujian dan analisis sistem untuk kemudian diambil kesimpulan.

3.1 Studi Literatur

Studi literatur adalah tahap untuk mengkaji dan mempelajari ilmu-ilmu yang berkaitan dengan penelitian “Sistem Deteksi Perhatian Operator Kamera Pengawas Terhadap Monitor Menggunakan Haar-cascade classifier” dari beberapa sumber pustaka seperti buku, jurnal, internet dengan sumber yang valid, dan penelitian sebelumnya yang terkait. Literatur yang terkait dengan penelitian ini adalah:

1. Penelitian mengenai perhatian operator pengawas kamera
2. Pengolahan citra digital menggunakan HSV dan Thresholding
3. Algoritma Haar-cascade classifier

3.2 Rekayasa Kebutuhan

Rekayasa kebutuhan bertujuan untuk mencatat kebutuhan yang diperlukan dalam penelitian ini. Rekayasa kebutuhan dilakukan dengan mengidentifikasi kebutuhan penelitian secara keseluruhan. Kebutuhan sistem dapat berupa kebutuhan Fungsional dan kebutuhan Non-fungsional. Diharapkan dengan adanya analisis kebutuhan, dapat membantu dan mempermudah perancangan sampai pembuatan sistem.

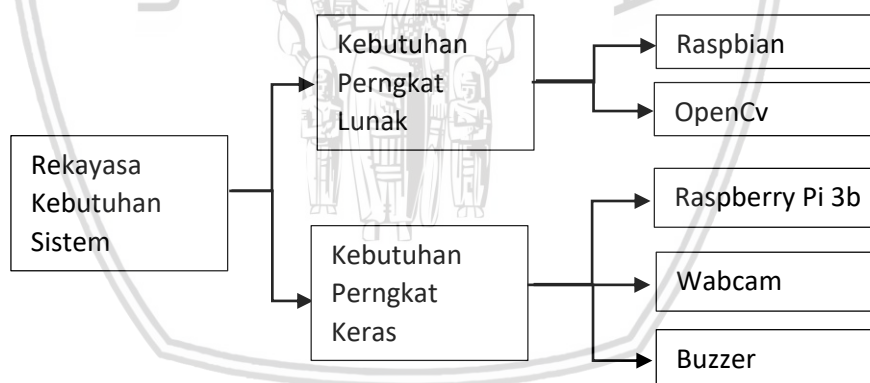
Kebutuhan perangkat lunak pada sistem adalah sebagai berikut:

1. Perangkat lunak sebagai perantara antara mikrokontroller dan manusia. Dimana perangkat lunak digunakan untuk mengatasi masalah pemrograman.

Kebutuhan perangkat keras pada sistem yaitu:

1. Mikrokontroller sebagai tempat memrogram dan sebagai pembaca program yang telah dibuat melalui perangkat lunak. Mikrokontroller juga digunakan sebagai pembaca input dari sensor.
2. Sensor yang digunakan sebagai pemberi input pada mikrokontroller untuk kemudian diproses.
3. Keluaran dari sistem berupa buzzer.

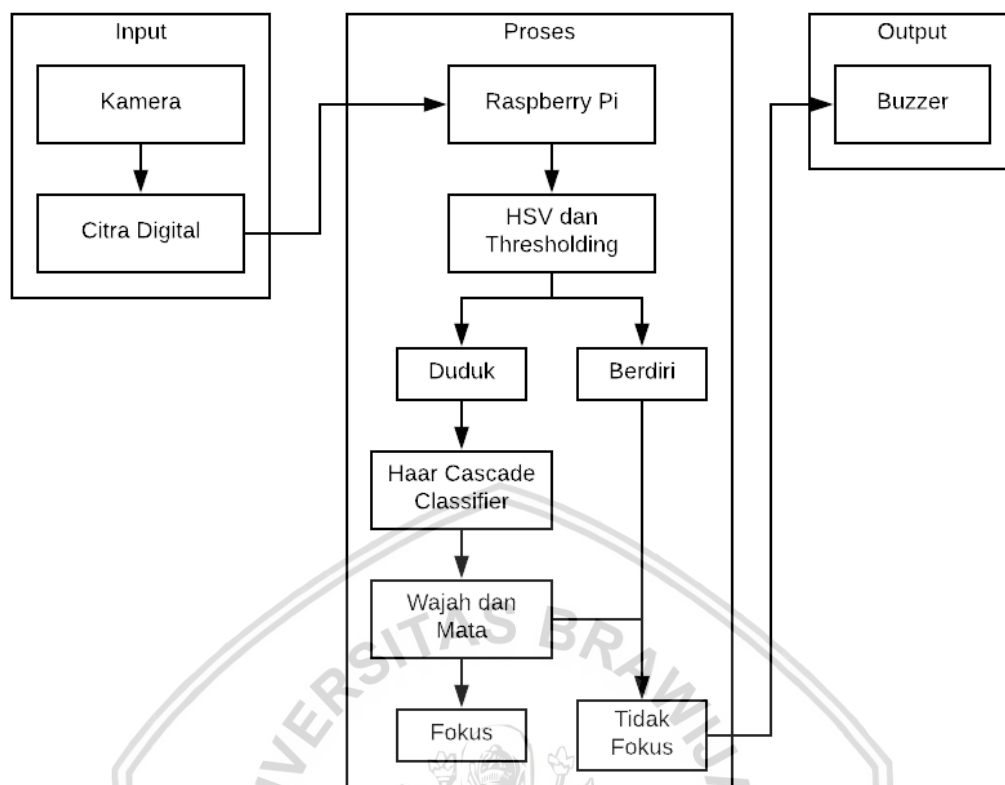
Berdasarkan rekayasa kebutuhan yang telah disebutkan sebelumnya yaitu kebutuhan perangkat lunak dan kebutuhan perangkat keras, rincian dari kebutuhan tersebut akan dijelaskan dengan pohon analisis pada Gambar 3.2 sebagai berikut.



Gambar 3.2 Bagam rekayasa kebutuhan

3.3 Perancangan Sistem

Tahap perancangan sistem memuat gambaran sistem yang akan dibuat dilihat dari arsitektur dan proses kerjanya menggunakan perangkat keras dan perangkat lunak yang telah dijelaskan pada rekayasa kebutuhan. Dengan membuat perancangan secara sistematis pada perangkat keras, dan perancangan perangkat lunak pada mikrokontroller, maka dapat dilakukan penyesuaian perangkat lunak dan perangkat keras pada sistem agar berjalan sesuai tujuan penelitian. Berikut ini adalah diagram blok perancangan sistem pada Gambar 3.3.



Gambar 3.3 Blok diagram desain sistem

Pada Gambar 3.3 dijelaskan bahwa input sistem berasal dari kamera. Kamera akan menangkap gambar operator secara real-time, dan mengirimkan data gambar ke Raspberry Pi. Dalam Raspberry Pi, akan dilakukan deteksi posisi (duduk atau berdiri) menggunakan HSV dan Thresholding. Jika gambar dideteksi sebagai posisi duduk, maka akan dilanjutkan deteksi wajah dan mata menggunakan algoritma Haar-cascade classifier. Data posisi yang didapat akan digunakan untuk klasifikasi tingkat perhatian operator. Hasil kategori akan menentukan apakah buzzer berbunyi (jika operator tidak memperhatikan monitor) atau tidak berbunyi (jika operator sedang memperhatikan monitor). Operator dikategorikan tidak memperhatikan minitor apabila dalam posisi berdiri atau memalingkan wajah saat duduk dan atau memejamkan mata saat duduk.

3.4 Implementasi

Pada tahap implementasi, akan dijelaskan tentang penerapan sistem berdasarkan analisis kebutuhan dan perancangan sistem yang telah dilakukan. Implementasi sistem dilakukan mulai dari perangkat keras dan perangkat lunak, berikut ini adalah tahapan implementasi:

1. Implementasi perangkat keras sistem deteksi perhatian operator. Penggabungan setiap komponen perangkat keras. Dirangkai sesuai dengan perancangan yang telah dilakukan.
2. Implementasi Pengolahan citra digital menggunakan HSV dan Thresholding untuk pendeteksi posisi duduk atau berdiri operator. Citra yang ditangkap oleh kamera akan langsung diproses dengan ruang warna HSV untuk

menyeleksi piksel dengan rentang warna kulit-manusia. Setelah mendapat piksel dengan warna kulit akan dithreshold dan dihitung jumlah piksel warna putihnya.

3. Implementasi algoritma Haar-cascade classifier untuk deteksi wajah dan mata manusia. Penerapan algoritma haar cascade untuk deteksi wajah manusia secara *real-time*. Jika wajah telah dideteksi, maka akan dilanjutkan untuk deteksi mata.
4. Implementasi klasifikasi perhatian
5. Implementasi keluaran sistem berupa buzzer.

3.5 Pengujian dan analisis Sistem

Tahap pengujian dan analisis dilakukan untuk menunjukkan tingkat keberhasilan sistem dalam menentukan kondisi memperhatikan monitor operator kamera pengawas dari perilaku yang ditunjukkan oleh operator. Tahap pengujian dan analisis meliputi:

1. Pengujian dan analisis akurasi deteksi posisi duduk dan berdiri operator menggunakan HSV dan Tresholding
2. Pengujian dan analisis akurasi deteksi wajah dan mata manusia menggunakan Haar-cascade classifier

3.6 Pengambilan Kesimpulan

Pengambilan kesimpulan dapat dilakukan setelah proses perancangan, implementasi, pengujian dan analisis sistem dilakukan. Kesimpulan diambil berdasarkan hasil pengujian sistem dan analisis sistem. Pada bagian kesimpulan diharapkan dapat menjadi acuan untuk mengembangkan penelitian ini dan memperbaiki kesalahan-kesalahan yang terjadi.

Selain kesimpulan, akan dibahas saran untuk penelitian yang telah dibuat. Saran berisi masukan yang dapat digunakan untuk mengembangkan penelitian deteksi perhatian operator. Saran juga dapat memberi ide kepada pembaca untuk melakukan penelitian yang berhubungan dengan perangkat keras dan metode yang digunakan di penelitian ini.

BAB 4 REKAYASA KEBUTUHAN

4.1 Deskripsi Umum

Bab ini membahas secara detail mengenai rekayasa kebutuhan yang wajib terdapat pada sistem untuk melanjutkan penelitian ke tahap implementasi sistem. Rekayasa kebutuhan diperlukan agar sistem dapat bekerja sesuai tujuan pembuatan sistem.

4.1.1 Perspektif Sistem

Sistem deteksi perhatian operator kamera pengawas terhadap monitor menggunakan algoritma Haar-cascade classifier merupakan sebuah alat yang mampu mendeteksi kondisi perhatian subjek pada monitor. Sistem ini dibuat untuk mengawasi kinerja operator pengawas kamera. Sistem akan mengolah input berupa citra digital dari subjek, kemudian sistem akan menentukan apakah subjek berada dalam kondisi memperhatikan monitor atau tidak memperhatikan monitor. Sistem ini menggunakan kamera sebagai sumber input, raspberry pi sebagai mikrokontroler pemroses input dan buzzer digunakan sebagai output.

4.1.2 Ruang Lingkup

Ruang lingkup sistem deteksi perhatian operator adalah penggunaan kamera sebagai sensor dan raspberry pi sebagai mikrokontroler pengolah input berupa citra digital. Hasil keluaran sistem adalah buzzer yang akan berbunyi jika operator kamera pengawas dalam kondisi tidak memperhatikan monitor. Berikut rincian dari ruang lingkup:

1. Ruang Lingkup Sensor

Ruang lingkup sensor pada sistem ini berupa sebuah kamera Logitech bertipe C270. Kamera ini beroperasi pada tegangan 5v dan arus 500 mA. Kamera ini digunakan untuk menangkap citra sejauh 40cm sampai 100cm untuk digunakan sebagai input.

2. Ruang Lingkup Output

Ruang lingkup output pada sistem ini berupa buzzer yang akan berbunyi jika kondisi operator pengawas kamera sedang tidak memperhatikan monitor.

4.1.3 Karakteristik Pengguna

Sistem ini merupakan prototype dan pengujian dilakukan oleh penulis. Pengguna sistem deteksi perhatian ini adalah pengguna tunggal dikarenakan sistem berupa prototype.

4.1.4 Lingkungan Operasi Sistem

Lingkungan operasi sistem yang mendukung berjalannya sistem ini adalah ruangan dengan pencahayaan ± 350 lux menurut SNI 03-6575-2001. Dimana SNI 03-6575-2001 membahas mengenai pencahayaan ruangan. Sebagai pendukung untuk pengambilan data pengujian pada prototype ini, maka ruangan yang digunakan adalah ruangan dengan cat dinding polos tidak berpola.

4.1.5 Batasan Perancangan dan Implementasi

Berikut ini adalah batasan perancangan dan implementasi pada sistem:

1. Sistem dapat mendeteksi subjek maksimal sebanyak satu subjek
2. Latar belakang subjek tidak memiliki unsur warna kulit manusia
3. Subjek harus berada dalam jangkauan kamera, jarak minimal subjek dari kamera sebesar $\pm 40\text{cm}$ dan maksimal $\pm 100\text{cm}$
4. Kondisi pencahayaan ruangan untuk implementasi disesuaikan dengan standar nasional Indonesia mengenai pengaturan pencahayaan ruangan yaitu sekitar 200 lux sampai 350lux
5. Sistem deteksi tingkat perhatian operator merupakan sebuah prototype yang belum dapat diterapkan pada kondisi sesungguhnya

4.1.6 Asumsi dan Ketergantungan

Dalam penerapan sistem terdapat asumsi dan ketergantungan, berikut ini adalah asumsi dan ketergantungan dari sistem:

1. Diasumsikan subjek ujicoba adalah operator pengawas kamera
2. Untuk mendapatkan citra digital yang bagus dari kamera maka perlu mengatur pencahayaan ruangan agar tidak terlalu redup atau terlalu terang
3. Untuk mendeteksi posisi duduk dan berdiri dengan baik, arah jangkauan kamera perlu diatur menyesuaikan tinggi subjek
4. Pengujian dilakukan pada setiap subjek sebanyak satu sesi selama ± 1 jam
5. Subjek yang diuji pada setiap sesi adalah satu subjek setiap sesi

4.2 Rekayasa Kebutuhan

Rekayasa kebutuhan menjelaskan mengenai kebutuhan dari sistem secara keseluruhan dan terperinci supaya sistem dapat beroperasi dengan baik dan sesuai dengan perancangan yang telah dibuat. Bagian ini membahas kebutuhan fungsional dan kebutuhan non fungsional sistem.

4.2.1 Kebutuhan Fungsional

Kebutuhan fungsional yaitu kebutuhan yang bersifat mutlak atau wajib dipenuhi agar tujuan dibuatnya sistem ini dapat terwujud. Kebutuhan tersebut dapat mengenai sesuatu yang dapat dilakukan oleh sistem ataupun hasil apa yang didapat dari sistem. Berikut adalah kebutuhan fungsional dari sistem:

1. Sistem dapat mengambil gambar dari kamera
 - a. Penjelasan dan Prioritas

Input didapat dari video yang ditangkap oleh kamera. Kamera menangkap video dan dikirim ke raspberry pi, dari video tersebut diproses setiap framenya untuk ditentukan posisi subjek. Fungsi ini memiliki prioritas tinggi. Fungsi ini menjadi sumber data input untuk diproses pada mikrokontroler.

b. Respon Sistem

Ketika kamera dihubungkan dengan mikrokontroller yang telah terhubung ke catu daya dan mikrokontroller, maka kamera dapat mulai beroperasi mengambil gambar untuk mendapatkan input yang dibutuhkan. Jika input telah didapatkan, maka sistem dapat melanjutkan ke proses pengolahan citra digital pada raspberry pi.

2. Sistem melakukan pemrosesan citra digital menggunakan Raspberry pi

a. Penjelasan dan Prioritas

Sistem ini mendapat data masukan dari kamera berupa citra digital, kemudian data tersebut diproses dengan HSV dan Thresholding dan Haar-cascade classifier. Proses ini akan dilakukan sampai sistem dapat memutuskan kondisi memperhatikan monitor atau tidak memperhatikan monitor. Prioritas dari fungsi ini tinggi karena berfungsi menentukan nilai keluaran dari sistem.

b. Respon Sistem

Sistem melakukan pemrosesan citra digital untuk mendeteksi posisi duduk atau berdiri menggunakan HSV dan Thresholding setelah mendapat *input* dari kamera. Proses ini dimulai dari mengubah gambar ke dalam ruang warna HSV untuk menyeleksi piksel dengan warna kulit. Piksel yang termasuk ke dalam rentang warna kulit akan di Threshold menjadi putih, sedangkan yang tidak termasuk rentang warna kulit akan di Threshold menjadi hitam. Untuk menentukan posisi duduk dan berdiri, frame akan dibagi menjadi dua yaitu atas dan bawah. Setelah membagi frame, setiap frame akan dihitung piksel putihnya. Jika frame bagian bawah terdapat lebih banyak piksel putih, maka subjek akan dikategorikan dalam keadaan duduk. Sebaliknya, jika frame bagian atas terdapat lebih banyak piksel putih, maka subjek akan dikategorikan dalam keadaan berdiri.

Setelah melakukan deteksi posisi duduk atau berdiri, jika subjek terdeteksi duduk maka sistem akan melanjutkan deteksi wajah dan mata dengan algoritma Haar-cascade classifier. Proses deteksi wajah dimulai dari konversi citra RGB menjadi *Gray* dengan perhitungan konversi RGB ke Luma pada ruang warna YCbCr. Gambar *gray* akan ditentukan *Haar-like feature*-nya untuk menemukan wajah menggunakan Adaboost *classifier*. Fitur *Haar-like* dihitung perbedaan nilai threshold-nya dengan perhitungan integral image.

Dari hasil deteksi posisi duduk atau berdiri dan deteksi wajah maka dapat dilakukan klasifikasi perhatian operator sebagai output sistem. Kondisi memperhatikan monitor terjadi apabila subjek dalam keadaan duduk dan wajah subjek beserta mata subjek terdeteksi.

3. Hasil keluaran dari sistem setelah pemrosesan raspberry pi adalah buzzer
 - a. Penjelasan dan Prioritas

Fungsi ini digunakan untuk menampilkan keluaran dari sistem dalam bentuk audio. Setelah melalui tahap klasifikasi kondisi, maka sistem akan memicu buzzer untuk bekerja. Buzzer akan berbunyi jika kondisi subjek sedang tidak memperhatikan monitor. Prioritas fungsi ini tinggi, untuk mengetahui apakah alat bekerja dengan baik.

- b. Respon Sistem

Setelah sistem memproses data input dan telah mengklasifikasi tingkat perhatian, maka sistem akan memicu buzzer untuk bekerja.

4.2.2 Kebutuhan Perangkat Keras

Kebutuhan perangkat keras sistem deteksi perhatian operator kamera pengawas terhadap monitor akan dijelaskan secara rinci pada Tabel 4.1.

Tabel 4.1 Kebutuhan fungsional perangkat keras

Kebutuhan Perangkat Keras	Fungsi	Keterangan
Kamera Logitech C270	Sebagai sensor pengambil input berupa citra	Kamera ini dapat menangkap video dengan resolusi 1280 x 720 pixels sehingga cukup untuk mengambil gambar wajah pada jarak 40cm sampai 100cm untuk deteksi wajah
Raspberry Pi 3b	Mikrokontroler pemroses input dari kamera	Mikrokontroler ini memiliki prosesor Quad Core 1.2GHz Broadcom BCM2837 yang mendukung pemrosesan citra digital
Buzzer	Sebagai hasil keluaran dari proses sistem	Buzzer digunakan sebagai <i>output</i> karena rangsangan audio lebih diperhatikan oleh manusia daripada rangsangan visual

4.2.3 Kebutuhan Perangkat Lunak

Kebutuhan perangkat lunak sistem akan dijelaskan pada Tabel 4.2 berikut.

Tabel 4.2 Kebutuhan perangkat lunak

Kebutuhan Perangkat Lunak	Fungsi	Keterangan
Sistem Operasi Raspbian	Sebagai sistem operasi yang digunakan pada raspberry pi 3b	Sistem operasi ini dikembangkan berdasarkan perangkat keras raspberry Pi
OpenCV	Penunjang pemrograman pengolahan citra berupa library yang dipasang pada sistem operasi raspbian	OpenCV merupakan perangkat penunjang pemrosesan citra digital yang cukup lengkap



BAB 5 PERANCANGAN DAN IMPLEMENTASI

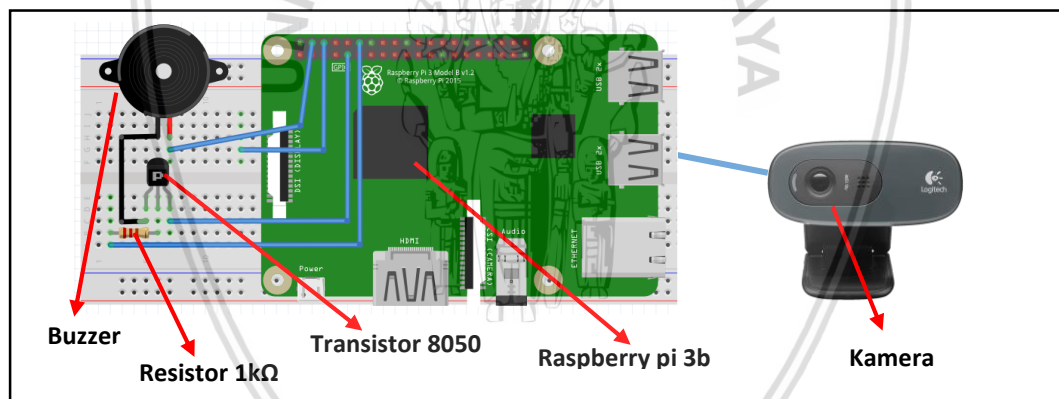
Bab ini menjelaskan perancangan dan implementasi pada penelitian ini. Perancangan membahas persiapan sebelum sistem di implementasikan. Implementasi membahas penerapan perancangan sistem sesuai dengan persiapan perancangan yang telah dibahas dan telah dilakukan sebelumnya.

5.1 Perancangan Sistem

Perancangan sistem membahas mengenai persiapan implementasi penelitian “Sistem Deteksi Perhatian Operator Kamera Pengawas Terhadap Monitor Menggunakan Haar-cascade classifier”. Perancangan sistem berupa perancangan perangkat keras dan perancangan perangkat lunak.

5.1.1 Perancangan Perangkat Keras

Perangkat keras yang akan digunakan pada penelitian ini dibagi menjadi tiga, yaitu input, pemroses, dan output. Input yang digunakan adalah citra digital yang didapat dari kamera Logitech C270. Pemroses pada sistem ini adalah raspberry pi tipe 3b sebagai *mikrokontroler*. Output dari sistem ini berupa bunyi dari buzzer. Perancangan perangkat keras dapat dilihat pada Gambar 5.1.



Gambar 5.1 Skema perancangan perangkat keras

Koneksi pin perancangan perangkat keras sistem deteksi perhatian operator terhadap monitor seperti pada Gambar 5.1 dijelaskan pada Tabel 5.1.

Tabel 5.1 Koneksi pin perancangan perangkat keras

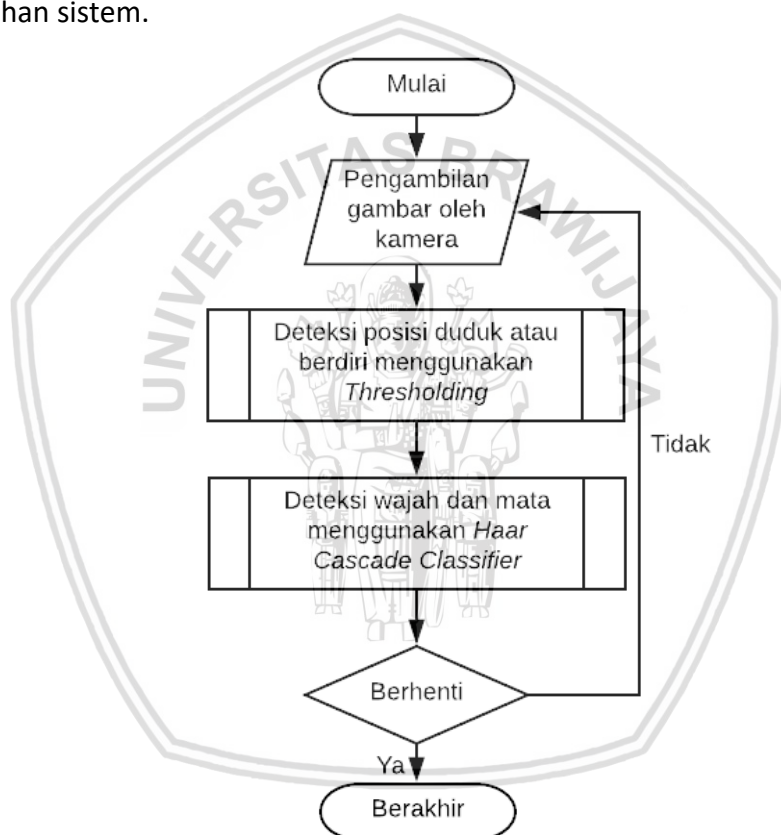
Pin dan Port Raspberry Pi	Kamera	Buzzer	PNP 8050	Resistor	Ground
Port USB 2.0	✓				
Pin 4 (5v)		Positif			
Pin 9 (Ground)			Emitter		
Pin 12				Kaki resistor	
		Negatif	Collector		
			Basis	Kaki resistor	
Pin 6 (Ground)					✓

Keterangan: Keterangan: ✓ =Terhubung

Berdasarkan perancangan pada Tabel 5.1 terdapat perangkat tambahan pada rangkaian buzzer. Perangkat tambahan tersebut berupa resistor $1k\Omega$ dan transistor 8050. Resistor pada rangkaian buzzer digunakan untuk menyetabilkan tegangan, sedangkan transistor digunakan sebagai *switch* yang akan menyambung atau memutus arus ke buzzer.

5.1.2 Perancangan Perangkat Lunak

Perancangan perangkat lunak pada sistem deteksi perhatian operator pengawas kamera terhadap monitor dilakukan dengan melakukan perhitungan algoritma dan mengerjakan kode dasar sebelum disatukan kedalam kode utama. Berikut ini adalah perancangan algoritma sistem deteksi perhatian operator pengawas kamera secara bertahap. Pada Gambar 5.2 adalah diagram alir proses keseluruhan sistem.

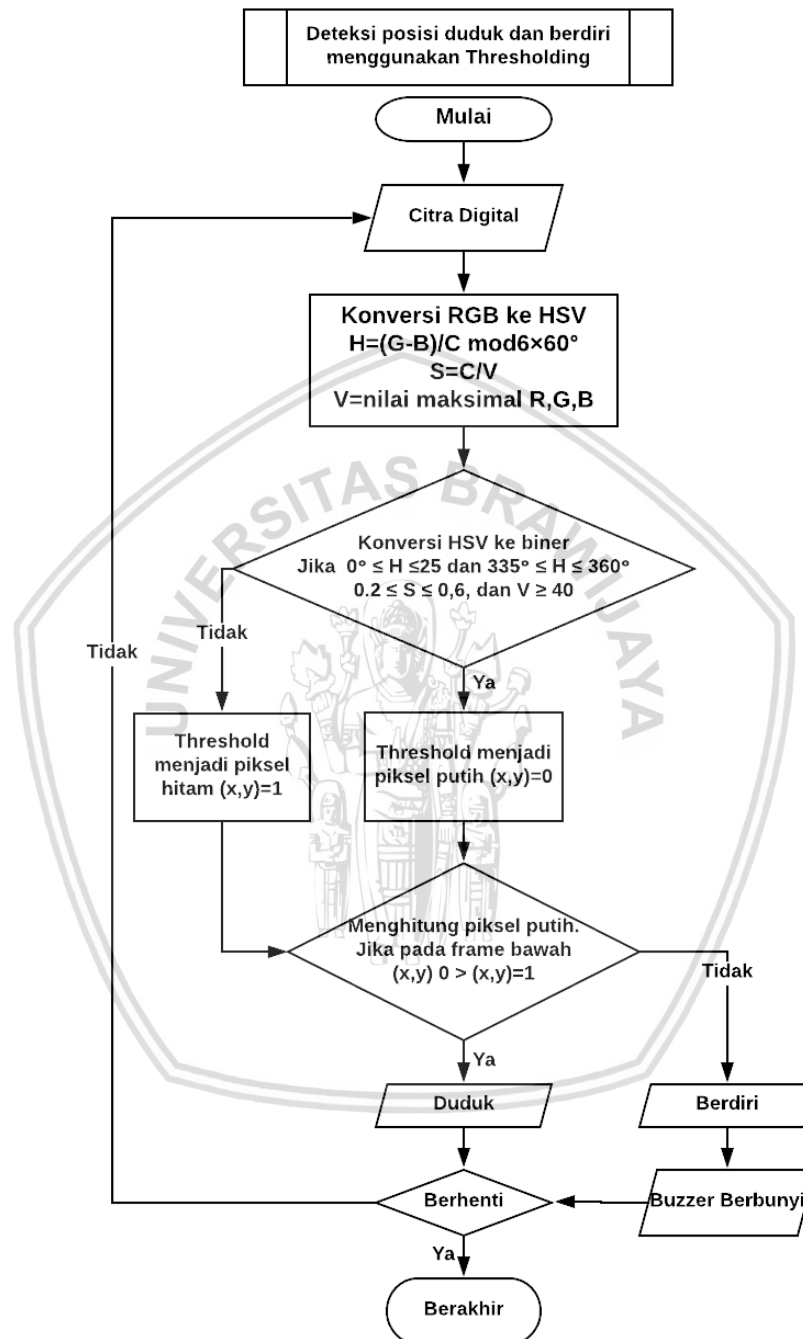


Gambar 5.2 Diagram alir proses sistem secara keseluruhan

Berdasarkan diagram alir pada Gambar 5.2 saat sistem dimulai, kamera akan mengambil gambar. Gambar yang didapat dari kamera akan di *thresholding* untuk mendeteksi posisi duduk dan berdiri. Setelah hasil posisi didapat, gambar akan dideteksi wajah dan mata menggunakan Haar Cascade Classifier. Hasil dari pengolahan citra menggunakan *Thresholding* dan algoritma Haar Cascade Classifier akan menentukan hasil keluaran buzzer. Jika program masih terus berjalan, proses diatas akan terus dilakukan sampai program dihentikan.

1. Perancangan pengolahan citra dengan HSV dan Tresholding

Proses pengolahan citra dengan HSV dan *Thresholding* untuk deteksi posisi duduk atau berdiri subjek akan dijelaskan pada Gambar 5.3 berikut.



Gambar 5.3 Cara kerja deteksi duduk atau berdiri dengan Thresholding

Berdasarkan Gambar 5.3, video yang didapat dari kamera dijadikan masukan untuk diolah dengan HSV dan Thresholding agar dapat mendeteksi posisi duduk dan berdiri. Konversi citra RGB ke HSV dilakukan untuk mendeteksi kulit manusia pada setiap frame. Deteksi kulit dilakukan dengan cara mengambil piksel yang berada

pada rentang warna kulit manusia. Pixel yang termasuk rentang warna kulit manusia dirubah menjadi pixel putih dan selain itu diubah menjadi hitam menggunakan Threshold.

Setelah proses threshold citra, akan dihitung jumlah pixel warna putih pada frame atas dan bawah untuk menentukan posisi. Jika pixel warna putih pada frame bagian bawah lebih banyak daripada bagian atas, maka akan disimpulkan bahwa operator sedang duduk dan sebaliknya adalah berdiri. Keluaran yang didapat setelah pemrosesan gambar dengan metode ini adalah posisi duduk atau berdiri dari subjek. Berikut ini adalah perhitungan HSV dan Tresholding untuk mendapatkan posisi duduk dan berdiri subjek.

Menurut Ilona Usuman, rentang warna pixel kulit manusia dengan perbedaan ruang warna HSV didapat Hue sebesar 0.021 sampai 0.080 dan Nilai Saturation sebesar 0.23 sampai 0.68 (Usuman, et al., 2012). Sedangkan Rentang warna pixel kulit manusia dengan perbedaan ruang warna HSV menurut Rahman yaitu $0^\circ \leq H \leq 25^\circ$ dan $335^\circ \leq H \leq 360^\circ$, $0.2 \leq S \leq 0.6$, dan $V \geq 40$ (Rahman, et al., 2014)

Sedangkan untuk menentukan rentang warna kulit menurut ruang warna HSV diperlukan nilai RGB dalam ruang warna RGB. Menurut Kolkur dalam penelitiannya, rentang warna kulit dalam ruang warna RGB didapat $R > 95$, $G > 40$, and $B > 20$ dimana $R > G$ dan $R > B$ (Kolkur, et al., 2016). Berikut ini adalah contoh perhitungan konversi warna RGB ke HSV dari sebuah pixel menggunakan standar warna kulit RGB.

Diketahui jika $R=185$, $G=124$, dan $B=109$ dimana $R > G$ dan $R > B$ maka:

$$M = 185 \quad m = 109 \quad C = 76$$

$$H' = \frac{124-109}{76} \bmod 6 = 0.19$$

$$H = 60^\circ \times 0.19 = 11.4^\circ$$

$$V = 185 \quad S = \frac{76}{185} = 0.41$$

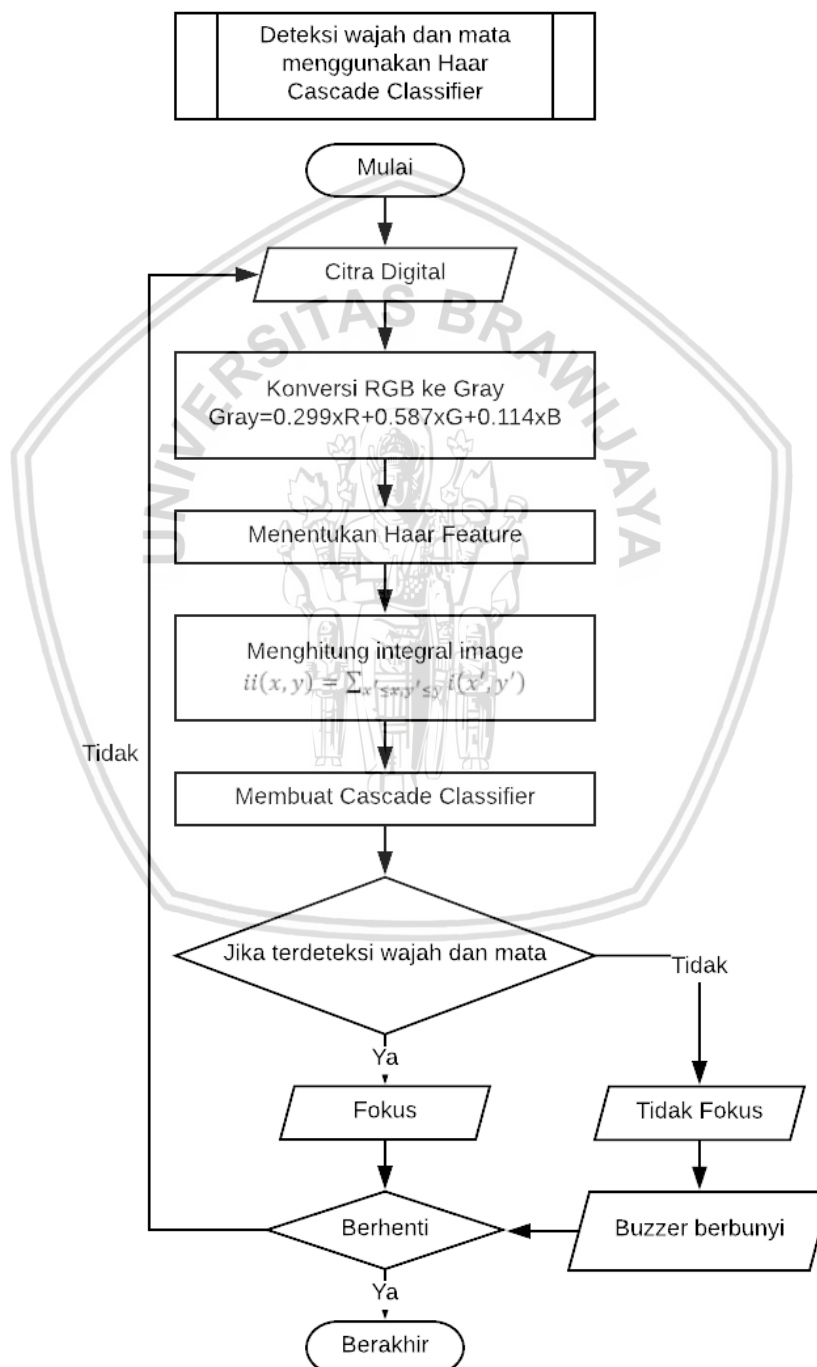
Setelah mendapatkan nilai Hue, Saturation, dan Value jika pixel tersebut termasuk kedalam range kulit manusia, maka selanjutnya adalah mengubah pixel tersebut menjadi putih dan pixel selain warna kulit menjadi hitam. Dimana rentang warna kulit yang digunakan pada penelitian ini yaitu $0^\circ \leq H \leq 25^\circ$ dan $335^\circ \leq H \leq 360^\circ$, $0.2 \leq S \leq 0.6$, dan $V \geq 40$ (Rahman, et al., 2014)

$$\text{Nilai biner } (x, y) = \begin{cases} 0, & \text{jika } I(x, y) \text{ memenuhi nilai HSV kulit. dan} \\ 1, & \text{jika } I(x, y) \text{ tidak memenuhi nilai HSV kulit} \end{cases}$$

Setelah binerisasi pixel dengan warna kulit, selanjutnya adalah deteksi posisi duduk dan berdiri dengan membagi frame menjadi atas dan bawah, lalu menghitung pixel berwarna putih pada frame atas dan bawah. Jumlah pixel berwarna putih akan menentukan apakah subjek dalam posisi duduk atau berdiri. Pada sistem ini, subjek akan terdeteksi berdiri jika pixel putih pada frame atas jumlahnya lebih banyak daripada jumlah pixel putih di frame bawah. Sebaliknya, jika jumlah pixel putih pada frame bawah lebih banyak daripada frame atas maka subjek dikategorikan dalam posisi duduk.

2. Perancangan pengolahan citra dengan Haar-cascade classifier

Setelah mendeteksi posisi duduk dan berdiri, jika subjek terdeteksi dalam posisi duduk maka selanjutnya akan dideteksi wajahnya. Wajah subjek dideteksi dengan algoritma Haar-cascade classifier. Keluaran yang didapat setelah pemrosesan gambar dengan metode ini adalah wajah subjek yang menghadap depan atau ke lain arah. Jika wajah subjek terdeteksi, maka proses selanjutnya adalah deteksi kedua mata subjek dengan Haar-cascade classifier. Tahap deteksi wajah dan mata akan dijelaskan pada gambar 5.4 berikut



Gambar 5.4 Proses deteksi wajah menggunakan *Haar cascade classifier*

a. Tahap Grayscale

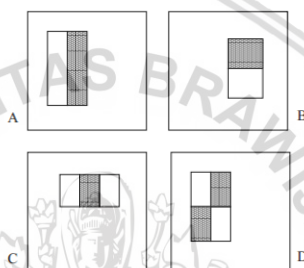
Tahap ini adalah tahap untuk mengubah citra dengan ruang warna RGB menjadi abu-abu. Dimana nilai R adalah nilai merah dari sebuah piksel, nilai G adalah nilai hijau dari sebuah piksel, dan nilai B adalah nilai biru dari sebuah piksel. Berikut ini adalah contoh konversi ruang warna RGB ke *Grayscale*, jika $R=100$, $G=50$, dan $B=10$ maka:

$$W = (0,2989 \times 100) + (0,5870 \times 50) + (0,1140 \times 10) = 60.38$$

Setelah gambar RGB dikonversi ke *grayscale* melalui ruang warna YCbCr, maka selanjutnya adalah menentukan *Haar feature*.

b. Menentukan Haar Feature

Fitur yang digunakan untuk proses klasifikasi gambar berdasarkan nilai dari fiturnya yaitu *Haar-like features*. Berikut ini adalah contoh fitur yang akan digunakan untuk deteksi wajah dan mata.



Gambar 5.5 Fitur kotak untuk mendeteksi wajah dan mata

Gambar 5.5 menunjukkan bahwa (A) dan (B) adalah fitur dua kotak. Sedangkan (C) dan (D) adalah fitur tiga kotak. Cara menghitung jumlah nilai dari fitur yang ada, yaitu dengan mengurangkan nilai piksel yang terdapat pada area putih dengan nilai piksel yang terdapat pada area hitam menggunakan perhitungan *integral image*.

c. Menghitung Integral Image

Proses menghitung *integral image* yaitu dengan menjumlahkan nilai piksel dari kiri atas sampai kanan bawah. Fitur kotak dapat dihitung dengan cepat menggunakan persamaan 5.1 berikut ini.

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y') \quad (5.1)$$

A	$s(x-1, y-1)$	1	B	$s(x, y-1)$	2
C	$s(x-1, y)$	3	D	$s(x, y)$	4

Gambar 5.6 Penerapan perhitungan pada fitur kotak

Berikut adalah penjelasan dari Gambar 5.6, dimana jumlah piksel kotak D dapat dihitung dengan empat referensi array. Nilai dari integral image pada area 1 adalah jumlah dari piksel pada kotak A, nilai pada area 2 adalah kotak

A+B, nilai pada lokasi 3 adalah A+C, dan lokasi 4 adalah A+B+C+D. jumlah dalam kotak D dapat dihitung dengan $4+1-(2+3)$. Jika $s(x, y)$ adalah jumlah baris kumulatif, maka:

$$s(x, y) = i(x, y) + s(x - 1, y) + s(x, y - 1) - s(x - 1, y - 1) \quad (5.2)$$

Dimana pada persamaan 5.2, $(x - 1)$ dan $(y - 1)$ adalah piksel tetangga dari piksel yang akan dihitung. Dengan operasi diatas, gambar integral bisa dihitung sekali operasi dari gambar asli. Dengan menggunakan integral image, jumlah dari daerah kotak dapat dihitung menggunakan empat array. Pada Gambar 5.7 berikut ini adalah contoh perhitungan integral image untuk menghitung jumlah fitur.

2	3	5	Maka:	2	2+3=5	2+3+5=10
3	5	4		2+3=5	2+3+3+5=13	2+3+5+3+5+4=22
2	4	3		2+3+2=7	2+3+5+3+2+4=19	2+3+5+3+5+4+2+4+3=31
1	2	1		2+3+2+1=8	2+3+5+3+4+2+1+2=22	2+3+5+3+5+4+2+4+3+1+2+1=35
2	3	2		2+3+2+1+2=10	2+3+3+5+2+4+1+2+2+3=27	2+3+5+3+5+4+2+4+3+1+2+1+2+3+2=42

Gambar 5.7 Contoh perhitungan integral image

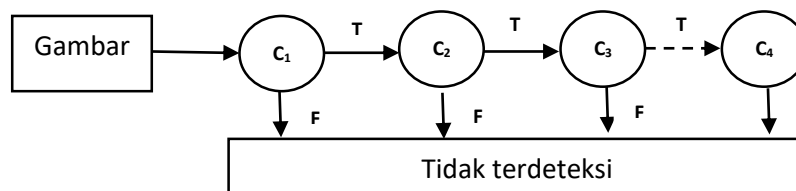
Dari Gambar 5.7 diatas didapat Fitur nilai Haar $D=D+A-(B+C)$

$$D = (10+0-0-2)-(27+2-10-5)+(42+5-27-10)$$

$$D = 8-14+10 = 4$$

d. Membuat Cascade Classifier

Untuk membuat cascade classifier menggunakan *machine learning* bernama Adaboost. Metode ini menggabungkan banyak *classifier* yang lemah untuk membangun *classifier* yang kuat. Classifier yang lemah yaitu urutan filter pada classifiernya hanya mendapat sedikit hasil benar.



Gambar 5.8 Proses yang dilalui gambar untuk mendeteksi wajah

Berdasarkan Gambar 5.8, klasifikasi pada tingkat pertama dimana setiap subgambar digolongkan menggunakan satu fitur, lalu menghasilkan nilai $T(true)$ jika gambar memenuhi fitur Haar tertentu dan $F(false)$ jika tidak

memenuhi fitur. Klasifikasi pertama akan mengurangi setengah beban klasifikasi pada tahap dua untuk diklasifikasikan. Hasil dari klasifikasi tahap dua berupa nilai T (*True*) jika gambar memenuhi proses integral image dan F (*false*) jika tidak memenuhi. Dengan bertambahnya tingkat klasifikasi, maka diperlukan syarat yang lebih rinci dari sebelumnya sehingga fitur yang akan digunakan menjadi lebih banyak dan bervariasi. Jumlah subgambar yang lulus klasifikasi akan semakin berkurang. Hasil klasifikasi tingkat akhir berupa T (*True*) jika gambar memenuhi proses AdaBoost dan F (*false*) jika tidak memenuhi proses (Putro, et al., 2012).

3. Perancangan Klasifikasi Perhatian Terhadap Monitor

Klasifikasi perhatian merupakan proses terakhir yang akan dilakukan untuk mendapat hasil keluaran sistem. Proses klasifikasi ini membutuhkan data hasil pengolahan video dengan binerisasi citra dan metode Haar-cascade classifier. Data tersebut akan digunakan untuk menggolongkan subjek dalam keadaan perhatian atau tidak memperhatikan monitor. Subjek dikatakan perhatian apabila subjek terdeteksi duduk dan wajahnya menghadap kamera. Sedangkan untuk kondisi tidak memperhatikan monitor, subjek akan terdeteksi berdiri atau menghadap kearah lain. Pada Tabel 5.2 akan dijelaskan klasifikasi kondisi subjek.

Tabel 5.2 Ketentuan kondisi subjek memperhatikan monitor

No	Duduk atau berdiri	Deteksi mata	Perhatian mengarah ke monitor
1	Berdiri	-	Tidak memperhatikan monitor
2	Duduk	Tidak terdeteksi	Tidak memperhatikan monitor
3	Duduk	Terdeteksi	Memperhatikan monitor

4. Perancangan Keluaran Sistem Dengan Buzzer

Hasil keluaran yang didapat dari sistem ini adalah kondisi memperhatikan monitor atau tidak memperhatikan monitor subjek yang ditunjukkan dengan bunyi dari buzzer. Jika subjek terdeteksi perhatian, maka buzzer akan diam. Sebaliknya, jika subjek terdeteksi tidak memperhatikan monitor maka buzzer akan berbunyi. Kondisi pengaktifan buzzer akan dijelaskan pada Tabel 5.3 berikut.

Tabel 5.3 Perancangan keluaran sistem dengan buzzer

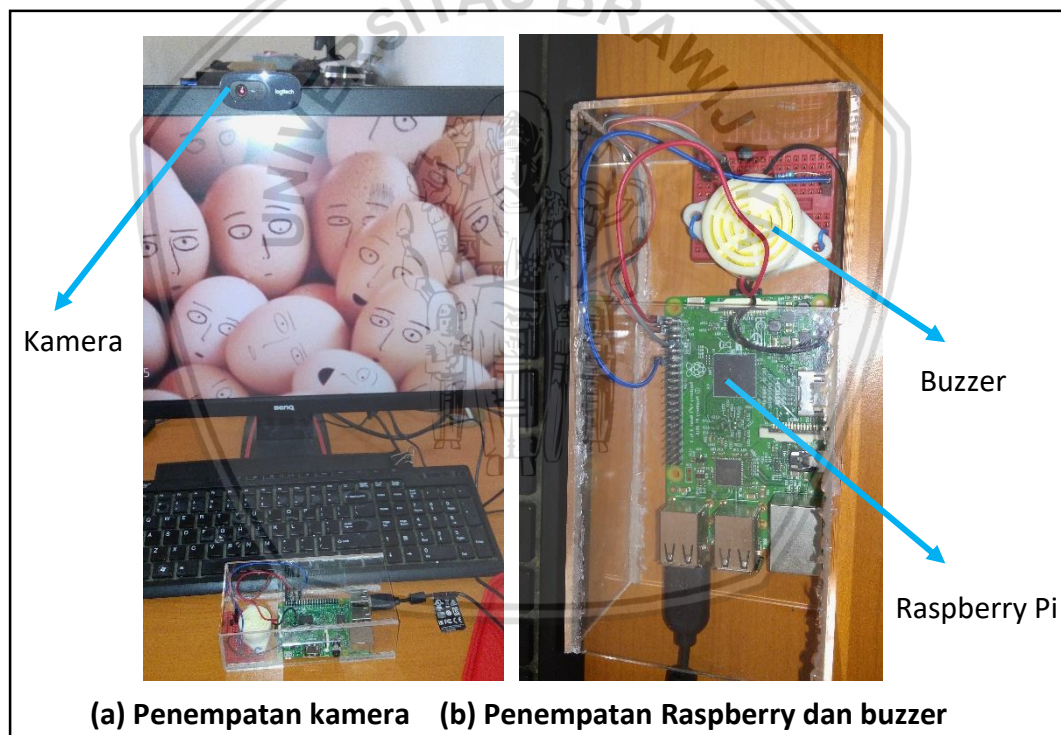
No	Posisi	Deteksi wajah dan mata	Perhatian mengarah ke monitor	Bunyi
1	Berdiri	-	Tidak memperhatikan monitor	Berbunyi
2	Duduk	Tidak terdeteksi	Tidak memperhatikan monitor	Berbunyi
3	Duduk	Terdeteksi	Memperhatikan monitor	Tidak Berbunyi

5.2 Implementasi Sistem

Pada implementasi sistem deteksi perhatian operator kamera pengawas terhadap monitoran dijelaskan mengenai penerapan sistem berdasarkan perancangan yang telah dibuat sebelumnya. Terdapat dua hal yang akan dibahas pada implementasi sistem, yaitu implementasi perangkat keras dan implementasi perangkat lunak.

5.2.1 Implementasi Perangkat Keras

Implementasi perangkat keras dilakukan berdasarkan perancangan perangkat keras yang telah dibuat sebelumnya. Sistem deteksi perhatian operator kamera pengawas terhadap monitor memiliki satu sumber masukan dan satu hasil keluaran. Masukan sistem ini berasal dari satu perangkat keras yaitu kamera. Perangkat pemroses masukan (mikrokontroler) dari kamera untuk menjadi keluaran yaitu Raspberry Pi. Sedangkan keluaran perangkat keras dari sistem ini menggunakan buzzer yang terhubung dengan mikrokontroler. Tampilan dari sistem secara keseluruhan dapat dilihat pada Gambar 5.6 berikut ini.



Gambar 5.9 Implementasi perangkat keras sistem

Implementasi bentuk sistem dari segi perangkat keras seperti pada Gambar 5.9 disesuaikan dengan perancangan sistem dengan tambahan tempat peletakan sistem. Perhatian utama implementasi sistem bergantung pada penempatan kamera sebagai sumber masukan sistem. Sistem ini menggunakan box agar lebih efisien dan dapat melindungi raspberry pi sekaligus rangkaian buzzer. Kamera yang digunakan untuk menangkap gambar diletakkan pada bagian atas layar subjek.

5.2.2 Implementasi Perangkat Lunak

Implementasi perangkat lunak pada sistem deteksi perhatian operator pengawas kamera dilakukan pada sistem operasi raspberry pi, yaitu raspbian yang telah terpasang opencv. Pada operasi raspbian, code dibuat pada file .cpp untuk dapat membaca citra digital, mengolahnya dan menentukan keluaran dari sistem. Agar kamera yang terhubung pada raspberry pi dapat menangkap gambar, diperlukan kode untuk memanggil fungsi kamera agar aktif. Setelah mendapat gambar digital, kode berikutnya digunakan mengolah citra digital untuk mengetahui kondisi yang dibutuhkan sebagai keluaran. Untuk keluaran sistem, kode pada raspberry pi dibuat untuk mengatur buzzer. Berikut ini adalah tahapan implementasi sistem deteksi perhatian operator pengawas kamera.

Implementasi sistem ini dibantu dengan *library* yang terdapat pada OpenCV. *Library* dapat dipanggil kedalam kode program dengan menyertakan *header files*, atau *include file* (pada kode C++). Header file digunakan untuk mempercepat waktu *compile* kode program dan menjaga agar kode program teteap terorganisir. Pada Tabel 5.4 adalah include file yang dibutuhkan agar sistem dapat berjalan.

Tabel 5.4 Include file yang dibutuhkan sistem deteksi perhatian operator

No	Kode Program
1	#include "opencv2/imgproc/imgproc.hpp"
2	#include "opencv2/highgui/highgui.hpp"
3	#include <opencv2/opencv.hpp>
4	#include <stdlib.h>
5	#include <stdio.h>
6	#include <iostream>
7	#include <opencv2/core/core.hpp>
8	#include <opencv2/objdetect/objdetect.hpp>
9	#include <wiringPi.h>

1. Implementasi Pengolahan citra digital menggunakan HSV dan Tresholding untuk pendeteksi posisi duduk atau berdiri operator.

Citra yang ditangkap oleh kamera berformat RGB diubah menjadi HSV. Setelah itu hasil konversi akan di *Thresholding*, dan ditentukan posisi duduk atau berdiri. Kode program untuk menentukan posisi duduk dan berdiri akan dijelaskan bertahap pada tabel-tabel berikut.

Tabel 5.5 Kode Program inialisasi variabel

No	Kode Program
1	int iLowH = 0;
2	int iHighH = 20;
3	int iLowS = 40;
4	int iHighS = 170;
5	int iLowV = 100;
6	int iHighV = 255;
7	int i,j, hitungatas, hitungbawah;
8	Mat frame, imgT,src_hsv;

Pada Tabel 5.5 adalah kode program untuk inialisasi variabel yang dibutuhkan untuk proses deteksi duduk dan berdiri. Dimana *iLow* adalah inialisasi nilai minimal dan *iHigh* adalah inialisasi nilai maksimal pada baris

1 sampai baris 6. Sedangkan huruf H, S dan V dibelakang *iLow* dan *iHigh* mewakili *Hue*, *Saturation*, dan *Value* dari ruang warna HSV. Inisialisasi pada baris 1 sampai baris 6 digunakan untuk menentukan rentang warna kulit yang digunakan untuk menyeleksi piksel. Mat pada baris 8 adalah class dengan dua bagian data, Mat dibutuhkan agar tidak perlu mengalokasikan memori secara manual dan melepaskan memori secara segera setelah tidak dibutuhkan.

Tabel 5.6 Kode program konversi *Thresholding*

No	Kode Program
1	<code>cvtColor(frame, src_hsv, CV_BGR2HSV);</code>
2	<code>inRange(src_hsv, Scalar(iLowH, iLowS, iLowV),</code>
3	<code>Scalar(iHighH, iHighS, iHighV), imgT);</code>
4	<code>erode(imgT, imgT, getStructuringElement(MORPH_ELLIPSE,</code>
5	<code>Size(5, 5)));</code>
6	<code>dilate(imgT, imgT, getStructuringElement(MORPH_ELLIPSE,</code>
7	<code>Size(5, 5)));</code>
8	<code>dilate(imgT, imgT, getStructuringElement(MORPH_ELLIPSE,</code>
9	<code>Size(5, 5)));</code>
10	<code>erode(imgT, imgT, getStructuringElement(MORPH_ELLIPSE,</code>
11	<code>Size(5, 5)));</code>

Pada Tabel 5.6 adalah kode program *Thresholding* hasil konversi gambar RGB ke HSV. Baris 1 digunakan untuk konversi RGB ke HSV menggunakan *library* opencv “CV_BGR2HSV”. *Library* ini dapat melakukan konversi citra dengan format warna RGB menjadi HSV. Gambar dengan format warna HSV diproses untuk menyeleksi piksel dengan rentang warna kulit manusia menggunakan *inRange* dan *Scalar* pada baris 3 dan 4. Setelah mendapat piksel dengan warna kulit, jika piksel termasuk dalam rentang warna kulit maka piksel tersebut akan dithreshold menjadi putih. Sebaliknya, piksel yang tidak termasuk dalam rentang warna kulit akan dithreshold menjadi warna hitam. Setelah mendapat gambar biner, dilakukan proses erosi dan dilatasi di baris 5 sampai 12 untuk membuang objek kecil dan untuk mengisi lubang akibat *noise*.

Tabel 5.7 kode program deteksi duduk dan berdiri

No	Kode Program
1	<code>for(i=0;i<imgT.cols;i++){</code>
2	<code>for(j=0;j<imgT.rows/2;j++){</code>
3	<code>int k=imgT.at<uchar>(j,i);</code>
4	<code>if(k==255){</code>
5	<code>hitungatas++; } }</code>
6	<code>for(i=0;i<imgT.cols;i++){</code>
7	<code>for(j=(imgT.rows/2)+1;j<imgT.rows;j++){</code>
8	<code>int k=imgT.at<uchar>(j,i);</code>
9	<code>if(k==255){</code>
10	<code>hitungbawah++; } }</code>
11	<code>if (hitungatas < hitungbawah){</code>
12	<code>cout<<"Duduk"<<" ";</code>
13	<code>}else cout<<"Berdiri"<<" ";</code>

Pada tabel 5.7 adalah kode program deteksi duduk dan berdiri dengan perulangan “for” dengan increment pada baris 1, 6 dan 7 untuk menghitung piksel putih. Perulangan pada baris 1 sampai 5 digunakan untuk menghitung

piksel putih pada frame bagian atas. Sedangkan baris 6 sampai 10 digunakan untuk menghitung piksel putih pada frame bagian bawah. Hasil perhitungan piksel putih digunakan untuk klasifikasi kondisi pada baris 11. Jika jumlah piksel putih sudah diketahui, selanjutnya adalah melakukan perbandingan jumlah untuk menentukan posisi duduk jika pada frame bawah terdapat lebih banyak piksel putih daripada frame atas. Jika piksel putih lebih banyak terdapat pada frame atas, maka program akan memberikan output “berdiri” pada terminal.

2. Implementasi algoritma Haar-cascade classifier untuk deteksi wajah dan mata.

Implementasi yang dilakukan berupa Penerapan algoritma *Haar cascade classifier* untuk deteksi wajah dan deteksi mata manusia secara *real-time*. Penerapan algoritma Haar Cascade menggunakan library yang terpasang pada openCV. Pada Tabel 5.8 adalah inisialisasi pemanggilan library yang digunakan untuk deteksi wajah dan mata manusia. Library yang digunakan adalah CascadeClassifier faceCascade untuk mengenali wajah manusia pada baris 3 sampai baris 4 dan CascadeClassifier eyeCascade untuk mengenali area mata manusia pada baris 8 sampai baris 10. Library OpenCv dapat dipanggil dengan menyertakan alamat *folder* library. Didalam library faceCascade dan eyeCascade terdapat proses untuk menentukan Haar Feature, menghitung Integral Image, dan membuat Cascade Classifier. Proses-proses tersebut yang akan dilalui untuk deteksi wajah dan mata.

Tabel 5.8 Kode untuk memanggil library deteksi wajah dan mata

No	Kode Program
1	<code>cv::CascadeClassifier faceCascade;</code>
2	<code>cv::CascadeClassifier eyeCascade;</code>
3	<code>if (!faceCascade.load("/home/pi/opencv-</code>
4	<code>2.4.13/data/haarcascades/haarcascade_frontalface_alt.xml"</code>
5	<code>)){</code>
6	<code>std::cerr << "Could not load face detector" << std::endl;</code>
7	<code>return -1; }</code>
8	<code>if (!eyeCascade.load("/home/pi/opencv-</code>
9	<code>2.4.13/data/haarcascades/haarcascade_eye_tree_eyeglasses.</code>
10	<code>xml")) {</code>
11	<code>std::cerr << "Could not load" << std::endl;</code>
12	<code>return -1; } }</code>

Pada Tabel 5.9 adalah kode program fungsi untuk deteksi mata dan wajah manusia. Sebelum dilakukan deteksi, pada baris 4 gambar dirubah ke YCbCr dengan mengambil komponen luma dari format warna YCbCr. Pada baris 5 gambar YCbCr diubah ke grayscale untuk mempermudah sistem menentukan Haar fitur. Proses deteksi wajah, memanggil library dari fungsi “faceCascade” yang telah diinisialisasi sebelumnya. Program mendeteksi objek dengan ukuran yang berbeda-beda pada gambar pada baris 8 dan baris 13 menggunakan `detectMultiScale`. Proses deteksi mata pada baris 13 memanggil library dari fungsi “eyeCascade”. `CV_HAAR_SCALE_IMAGE` pada baris 9 dan baris 14 digunakan untuk downscale gambar. Setelah mendapatk area wajah dan mata, program menandai area tersebut dengan kotak virtual di baris 19.

Tabel 5.9 Kode program untuk deteksi wajah dan mata

No	Kode Program
1	void detectEyes(cv::Mat &frame, cv::CascadeClassifier
2	&faceCascade, cv::CascadeClassifier &eyeCascade){
3	cv::Mat grayscale, ybcr;
4	cv::cvtColor(frame, ybcr, CV_BGR2YCRGB);
5	cv::cvtColor(ybcr, grayscale, CV_BGR2GRAY);
6	cv::equalizeHist(grayscale, grayscale);
7	std::vector<cv::Rect> faces;
8	faceCascade.detectMultiScale(grayscale, faces, 1.1, 2, 0
9	CV_HAAR_SCALE_IMAGE, cv::Size(150, 150));
10	if (faces.size() == 0) return;
11	cv::Mat face = grayscale(faces[0]);
12	std::vector<cv::Rect> eyes;
13	eyeCascade.detectMultiScale(face, eyes, 1.1, 2, 0
14	CV_HAAR_SCALE_IMAGE, cv::Size(30, 30));
15	rectangle(frame, faces[0].tl(), faces[0].br(),
16	cv::Scalar(255, 0, 0), 2);
17	if (eyes.size() != 2) return;
18	for (cv::Rect &eye : eyes){
19	rectangle(frame, faces[0].tl() + eye.tl(), faces[0].tl()
20	+ eye.br(), cv::Scalar(0, 255, 0), 2); }

3. Implementasi klasifikasi perhatian operator terhadap monitor

Setelah mendapatkan input yang dibutuhkan untuk diklasifikasi, berupa posisi duduk atau berdiri dan hasil deteksi wajah dan mata subjek maka selanjutnya adalah klasifikasi input. Input diklasifikasikan berdasarkan perancangan sistem yang telah dilakukan sebelumnya. Berikut ini adalah kode program untuk klasifikasi perhatian operator.

Tabel 5.10 Kode program untuk klasifikasi perhatian

No	Kode Program
1	int tf = 0;
2	int td = 0;
3	
4	void detectEyes(cv::Mat &frame, cv::CascadeClassifier
5	&faceCascade, cv::CascadeClassifier &eyeCascade){
6	if (faces.size() == 0) { td++;
7	if (td >10) {
8	cout<<"dan Tidak memperhatikan
9	monitor";
10	beep ();}return;
11	} else td = 0;
12	if (eyes.size() != 2){ tf++;
13	if(tf >= 5){
14	cout<<"dan Tidak memperhatikan
15	monitor";
16	beep (); }
17	} else {cout<<"dan Perhatian";
18	tf = 0;
19	}}
20	int main(int argc, char** argv){
21	if (hitungatas < hitungbawah) {
22	cout<<"Duduk"<<" ";
23	}else cout<<"Berdiri"<<" ";

Pada Tabel 5.10 adalah kode program klasifikasi perhatian yang dimulai dari inisialisasi variabel pada baris 1 dan baris 2. Kode klasifikasi perhatian akan diletakkan pada fungsi `detectEyes` di baris 4 dan pada fungsi `main` di baris 18. Pada baris 6 dan baris 12 terdapat `"td"` dan `"tf"` yang merupakan increment yang digunakan sistem untuk memberikan waktu deteksi. Angka 10 dan angka 5 yang terdapat pada kondisi `"td"` dan `"tf"` di baris 7 dan 13 merupakan jumlah *frame*. Pada fungsi `detectEyes`, kode klasifikasi menggunakan kondisi jika maka untuk menentukan kondisi memperhatikan monitor atau tidak memperhatikan monitor dengan deteksi wajah di baris 6 dan deteksi mata di baris 11. Sedangkan pada fungsi `main`, kode klasifikasi kondisi pada baris 21 digunakan untuk menentukan posisi duduk dan berdiri manusia menggunakan hasil perhitungan piksel putih pada frame atas dan frame bawah.

4. Implementasi hasil keluaran sistem

Hasil keluaran sistem deteksi perhatian kamera pengawas yaitu berupa bunyi buzzer. Buzzer akan berbunyi sesuai dengan hasil klasifikasi perhatian. Berikut ini adalah kode program untuk mengaktifkan buzzer sesuai hasil klasifikasi.

Tabel 5.11 Kode program untuk mengaktifkan fungsi buzzer

No	Kode Program
1	#define BEEP 1
2	void beep () {
3	digitalWrite(BEEP,HIGH);
4	delay (100);
5	digitalWrite(BEEP,LOW); }
6	
7	int main(int argc, char** argv){
8	if(wiringPiSetup() == -1){return 1;
9	} pinMode(BEEP, OUTPUT); }

Pada Tabel 5.11 dijelaskan bahwa fungsi `beep` pada baris 2 adalah inisialisasi kondisi keluaran buzzer, jika *high* (baris 3) maka buzzer akan berbunyi selama 100ms dan jika *low* (baris 5) maka buzzer akan mati. Pada fungsi `main` terdapat `wiringPiSetup()` di baris 8 untuk mengaktifkan pin GPIO raspberry pi, sedangkan kode `pinMode` pada baris 9 digunakan untuk mengatur mode pin sebagai *output*.

BAB 6 PENGUJIAN DAN ANALISIS SISTEM

Pengujian dan analisis akan membahas mengenai pengujian dan analisis hasil penelitian Sistem Deteksi Perhatian Operator Pengawas Kamera Menggunakan Haar Cascade Classifier. Pengujian dan analisis dapat dilakukan setelah melalui proses perancangan dan implementasi. Pengujian yang dilakukan pada sistem berupa pengujian akurasi deteksi posisi duduk dan berdiri dan pengujian akurasi deteksi wajah dan deteksi mata manusia.

6.1 Pengujian dan Analisis Akurasi Deteksi Posisi Duduk dan Berdiri Operator Menggunakan HSV Dan Tresholding

6.1.1 Tujuan

Tujuan dari pengujian akurasi deteksi posisi duduk dan berdiri operator adalah untuk mengetahui ketepatan sistem dalam mendeteksi posisi duduk atau berdiri subjek dari kamera dan mengetahui kesesuaian deteksi berdasarkan perancangan sistem. Hasil keluaran sistem ini akan memutuskan untuk memanggil proses deteksi wajah dan deteksi mata atau tidak. Pengujian akurasi deteksi duduk dan berdiri dilakukan pada 10 subjek. Subjek dihadapkan secara langsung pada kamera, pengujian dilakukan secara bergantian.

6.1.2 Alat

Alat yang digunakan untuk pengujian akurasi deteksi duduk dan berdiri operator pengawas kamera berupa:

1. Kamera
2. Raspberry pi 3 tipe B
3. Layar monitor
4. Keyboard dan Mouse

6.1.3 Metode

Metode yang dilakukan untuk melaksanakan pengujian deteksi duduk dan berdiri subjek yaitu:

1. Melakukan pemasangan kamera pada raspberry pi
2. Menempatkan dan megatur posisi kamera pada monitor
3. Menghubungkan raspberry pi dengan monitor, keyboard, mouse dan sumber daya
4. Menyalakan raspberry pi dan menjalankan program deteksi perhatian
5. Memberikan input berupa 10 orang subjek yang duduk dan berdiri didepan kamera secara bergantian
6. Mengamati dan mengambil data hasil keluaran sistem
7. Menentukan tingkat keberhasilan sistem dalam mendeteksi duduk dan berdiri subjek dengan persamaan 6.1:

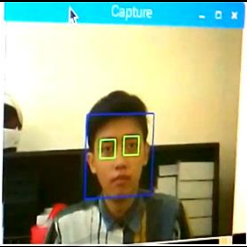

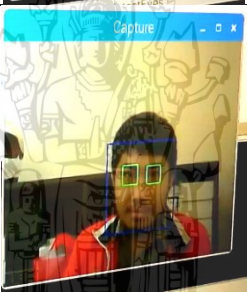


$$\text{Tingkat kesuksesan} = \frac{\text{jumlah berhasil}}{\text{Banyak percobaan}} \times 100\% \quad (6.1)$$

8. Menganalisis dan mengambil kesimpulan

6.1.4 Hasil

Gambar pengujian deteksi duduk dan berdiri dapat dilihat pada Tabel 6.1. Untuk hasil pengujian akurasi deteksi posisi duduk dan berdiri subjek berdasarkan Tabel 6.1 dapat dilihat pada Tabel 6.2 berikut.

Tabel 6.1 Hasil pengujian deteksi duduk dan berdiri

Pengujian	Duduk	Berdiri
Pengujian Subjek Kesatu		
Pengujian Subjek Kedua		
Pengujian Subjek Ketiga		
Pengujian Subjek Keempat		
Pengujian Subjek Kelima		

Pengujian	Duduk	Berdiri
Pengujian Subjek Keenam		
Pengujian Subjek Ketujuh		
Pengujian Subjek Kedelapan		
Pengujian Subjek Kesembilan		
Pengujian Subjek Kesepuluh		

Tabel 6.2 Hasil pengujian deteksi duduk dan berdiri subjek

No	Pengujian	Duduk	Berdiri
1	Pengujian Subjek Kesatu	Terdeteksi	Terdeteksi
2	Pengujian Subjek Kedua	Terdeteksi	Terdeteksi
3	Pengujian Subjek Ketiga	Terdeteksi	Terdeteksi
4	Pengujian Subjek Keempat	Terdeteksi	Terdeteksi
5	Pengujian Subjek Kelima	Terdeteksi	Terdeteksi
6	Pengujian Subjek Keenam	Terdeteksi	Terdeteksi
7	Pengujian Subjek Ketujuh	Terdeteksi	Terdeteksi
8	Pengujian Subjek Kedelapan	Terdeteksi	Terdeteksi
9	Pengujian Subjek Kesembilan	Terdeteksi	Terdeteksi
10	Pengujian Subjek Kesepuluh	Terdeteksi	Terdeteksi
Persentase keberhasilan		100%	100%

6.1.5 Analisis

Berdasarkan Tabel 6.1 yang menjelaskan gambar hasil pengujian deteksi duduk atau berdiri dan Tabel 6.2 yang menjelaskan keterangan hasil pengujian sistem terhadap posisi duduk atau berdiri subjek. Dari persentase hasil deteksi duduk dan berdiri, didapat rata-rata hasil akhir deteksi duduk dan berdiri sebesar 100% dengan jumlah subjek yang diuji 10 orang. Sistem menghasilkan rata-rata tingkat keberhasilan sebanyak 100% dari persentase deteksi posisi duduk sebanyak 100% dan berdiri sebanyak 100%. Dari 10 subjek yang diuji sistem, seluruh subjek terdeteksi dengan tepat. Walaupun sistem dapat mendeteksi dengan baik, tetapi sistem masih belum sempurna. Dikarenakan sistem mengenali posisi duduk dan berdiri subjek menggunakan perhitungan piksel putih yang terdapat pada frame, maka penempatan kamera menentukan akurasi deteksi.

6.2 Pengujian dan Analisis Akurasi Deteksi Wajah dan Deteksi Mata Manusia Menggunakan Haar-Cascade Classifier

6.2.1 Tujuan

Tujuan dari pengujian akurasi deteksi wajah dan mata manusia menggunakan Haar cascade classifier yaitu untuk mengetahui akurasi sistem dalam mendeteksi wajah dan mata subjek dan mata subjek dan mengetahui kesesuaian deteksi berdasarkan perancangan sistem. Hasil keluaran yang didapat setelah proses ini adalah perhatian dan tidak memperhatikan monitor subjek. Jika sistem berhasil mendeteksi wajah atau mata dengan baik, maka tujuan pengujian deteksi wajah dan mata telah tercapai. Pengujian dilakukan pada 10 subjek. Subjek dihadapkan secara langsung pada kamera, pengujian dilakukan secara bergantian.

6.2.2 Alat

Alat yang digunakan untuk pengujian akurasi deteksi wajah dan deteksi mata operator pengawas kamera berupa:

1. Kamera
2. Raspberry pi 3 tipe B
3. Layar monitor
4. Keyboard dan mouse

6.2.3 Metode

Metode yang dilakukan untuk melaksanakan pengujian deteksi wajah dan deteksi mata subjek yaitu:

1. Melakukan pemasangan kamera pada raspberry pi
2. Menempatkan dan megatur posisi kamera pada monitor
3. Menghubungkan raspberry pi dengan monitor, keyboard, mouse dan sumber daya
4. Menyalakan raspberry pi dan menjalankan program deteksi perhatian
5. Memberikan input didepan kamera berupa subjek dengan posisi:
 - a. Wajah menghadap depan
 - b. Wajah menoleh ke kanan











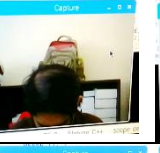



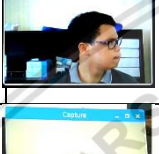
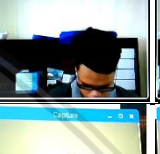
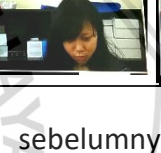
- c. Wajah menoleh ke kiri
- d. Wajah mendongak keatas
- e. Wajah menunduk kebawah
- f. Memejamkan mata
- g. Subjek maju setiap 20cm dari jarak 100cm mendekati kamera
- h. Kepala miring ke kanan 15°
- i. Kepala miring ke kiri 15°
6. Mengamati dan mengambil data hasil keluaran sistem
7. Menentukan tingkat keberhasilan sistem dalam mendeteksi wajah dan mata subjek dengan persamaan 6.1
8. Menganalisis dan mengambil kesimpulan

6.2.4 Hasil

Berikut pada Tabel 6.5 adalah Kesesuaian hasil klasifikasi sistem saat mendeteksi perhatian subjek terhadap monitor berdasarkan hasil dari Tabel 6.3. Untuk akurasi deteksi wajah dan deteksi mata pada saat kepala miring ditunjukkan pada Tabel 6.7 berdasarkan hasil dari Tabel 6.6. Hasil akurasi deteksi wajah dan deteksi mata pada jarak tertentu ditunjukkan pada Tabel 6.9 berdasarkan hasil dari Tabel 6.8.

Tabel 6.3 Gambar hasil deteksi wajah dan mata

No	Posisi hadap wajah					Mata tertutup
	Depan	Kanan	Kiri	Atas	Bawah	
1						
2						
3						
4						
5						

No	Posisi hadap wajah					Mata tertutup
	Depan	Kanan	Kiri	Atas	Bawah	
6						
7						
8						
9						
10						

Tabel 6.4 menunjukkan hasil perancangan sebelumnya mengenai klasifikasi sistem terhadap perhatian subjek ke monitor.

Tabel 6.4 Hasil Perancangan klasifikasi perhatian subjek terhadap monitor

No	Posisi	Deteksi wajah dan mata	Perhatian mengarah ke monitor	Bunyi
1	Berdiri	-	Tidak memperhatikan monitor	Berbunyi
2	Duduk	Tidak terdeteksi	Tidak memperhatikan monitor	Berbunyi
3	Duduk	Terdeteksi	Memperhatikan monitor	Tidak Berbunyi

Tabel 6.5 Kesesuaian hasil klasifikasi sistem saat mendeteksi perhatian subjek terhadap monitor

No	Subjek	Posisi hadap wajah					Mata tertutup
		Depan	Kanan	Kiri	Atas	Bawah	
1	Pertama	√	√	√	√	√	√
2	Kedua	√	√	√	√	√	√
3	Ketiga	√	√	√	√	√	√
4	Keempat	√	√	√	×	√	√
5	Kelima	√	√	√	√	√	√
6	Keenam	√	√	√	√	√	√
7	Ketujuh	√	√	√	√	√	√
8	Kedelapan	√	√	√	√	√	√

No	Subjek	Posisi hadap wajah					Mata tertutup
		Depan	Kanan	Kiri	Atas	Bawah	
9	Kesembilan	√	√	√	√	√	√
10	Kesepuluh	√	√	√	√	√	√
Persentase		100%	100%	100%	90%	100%	100%

Keterangan:

1. √ : Sesuai
2. × : Tidak sesuai

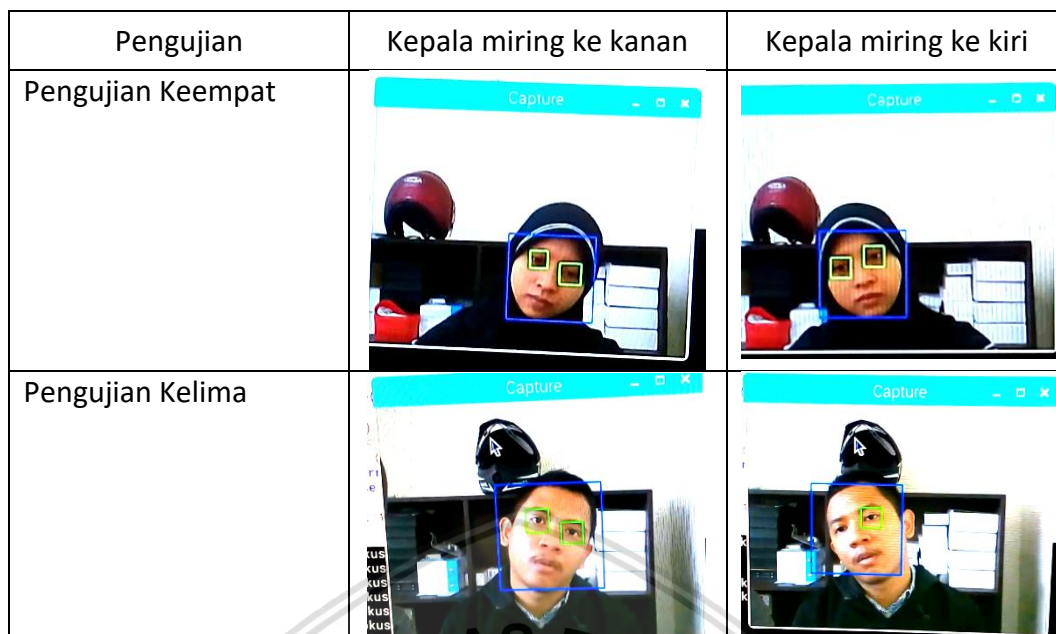
Berdasarkan gambar pada Tabel 6.3 dan hasil deteksi pada Tabel 6.5 didapat rata-rata keberhasilan sistem sebesar:

$$\text{Tingkat kesuksesan} = \frac{59}{60} \times 100\%$$

$$\text{Tingkat kesuksesan} = 98,33\%$$

Tabel 6.6 Gambar hasil deteksi wajah dan mata ketika kepala miring

Pengujian	Kepala miring ke kanan	Kepala miring ke kiri
Pengujian Pertama		
Pengujian Kedua		
Pengujian Ketiga		




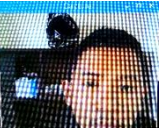







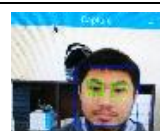
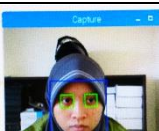





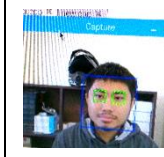
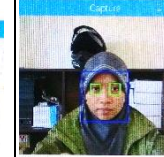
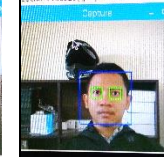

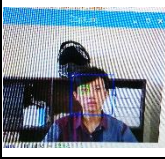

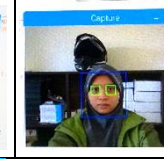
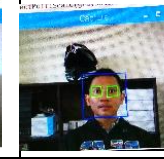
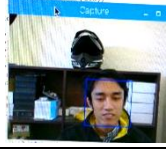




Berdasarkan dari gambar-gambar hasil deteksi wajah dan mata ketika kepala miring pada Tabel 6.6, maka didapat kesimpulan hasil deteksi wajah dan mata ketika kepala miring seperti pada Tabel 6.7.

Tabel 6.7 Hasil deteksi wajah dan mata ketika kepala miring

No	Subjek	Kepala miring ke kanan	Kepala miring ke kiri
1	Pertama	√	√
2	Kedua	√	√
3	Ketiga	√	√
4	Keempat	√	√
5	Kelima	√	√
Keberhasilan		100%	100%

Tabel 6.8 Gambar hasil akurasi jarak deteksi wajah dan deteksi mata

Jarak	Subjek				
	Pertama	Kedua	Ketiga	Keempat	Kelima
20cm					
40cm					
60cm					

Jarak	Subjek				
	Pertama	Kedua	Ketiga	Keempat	Kelima
80cm					
100cm					
120cm					

Berdasarkan dari gambar-gambar hasil deteksi wajah dan mata pada Tabel 6.7 yang dilakukan pada jarak 20cm sampai 120cm dengan rentang jarak 20cm pada setiap subjek maka didapat kesimpulan hasil deteksi wajah dan mata pada jarak tertentu seperti pada Tabel 6.8.

Tabel 6.9 Hasil akurasi jarak deteksi wajah dan deteksi mata

Jarak	Subjek				
	Pertama	Kedua	Ketiga	Keempat	Kelima
20cm	×	×	×	×	×
40cm	✓	✓	✓	✓	✓
60cm	✓	✓	✓	✓	✓
80cm	✓	✓	✓	✓	✓
100cm	✓	×	✓	✓	✓
120cm	×	×	×	×	×

Keterangan:

1. ✓ : Terdeteksi
2. × : Tidak terdeteksi

6.2.5 Analisis

Merujuk pada Tabel 6.3 dan Tabel 6.5 dapat dilihat bahwa sistem bekerja sesuai perancangan yang dilakukan sebelumnya. Dimana tingkat keberhasilan rata-rata dalam mendeteksi wajah dan mata untuk klasifikasi tidak memperhatikan monitor sebesar 98.33%. Keberhasilan sistem pada uji coba deteksi wajah dan mata untuk fungsi klasifikasi perhatian masih tergolong kurang akurat. Ketidakakuratan tersebut berupa terdeteksinya wajah dan mata saat wajah tidak berpaling melebihi 15° . Pada gambar ke empat dengan posisi wajah mendongak ke atas, wajah dan mata subek terdeteksi karena subjek tidak mendongak ke atas melebihi 15° .

Pada Tabel 6.6 dan Tabel 6.7 menunjukkan hasil akurasi sistem dalam mendeteksi wajah dan mata subjek dalam keadaan kepala miring ke kanan atau

ke kiri. Hasil rata-rata akurasi deteksi wajah dan mata saat kepala subjek miring ke kanan dan ke kiri sebesar 100%. Jika subjek memiringkan wajah tidak lebih dari 15° maka sistem dapat mendeteksi wajah dan mata dengan baik.

Berdasarkan Tabel 6.8 adalah gambar hasil pengujian jarak optimal sistem. Tabel 6.9 adalah hasil dari pengujian jarak optimal sistem agar dapat bekerja dengan baik. Pengujian dilakukan pada rentang jarak 20cm sampai 120cm dengan selisih jarak 20cm. Dari hasil pengujian tersebut, jarak minimum subjek dari kamera adalah ± 40 cm, sedangkan jarak maksimum subjek agar terdeteksi kamera adalah ± 100 cm. Jarak paling optimal subjek dari kamera adalah 40cm sampai dengan 80cm. Kamera dapat mendeteksi subjek jika jarak subjek tidak terlalu dekat dari kamera, karena deteksi wajah membutuhkan wajah subjek secara utuh dalam jangkauan *frame* kamera. Kamera juga tidak dapat mendeteksi subjek jika subjek terlalu jauh dari kamera, disebabkan oleh resolusi kamera sistem.



BAB 7 PENUTUP

Pada Bab penutup membahas mengenai kesimpulan yang diperoleh dari proses penelitian deteksi perhatian terhadap monitor yang telah dilakukan sebelumnya. Selain itu, penutup membahas mengenai saran untuk pengembangan penelitian Deteksi Perhatian Operator Kamera Pengawas Terhadap monitor.

7.1 Kesimpulan

Berdasarkan proses penelitian yang telah dilakukan, rumusan masalah yang dibuat sebelumnya dan hasil penelitian yang ada, maka dapat disimpulkan bahwa:

1. Proses membangun sistem deteksi perhatian menggunakan bunyi buzzer untuk memberi peringatan jika operator pengawas tidak memperhatikan monitor. Sistem peringatan menggunakan audio dinilai lebih mendapat perhatian daripada peringatan berbentuk visual. Kondisi tidak memperhatikan monitor operator ditunjukkan jika operator berdiri atau tidak berada pada jangkauan kamera dan jika operator tidak menghadapkan wajah pada area jangkauan kamera.
2. Pemrosesan citra menggunakan HSV dan *thresholding* dapat digunakan untuk menentukan posisi duduk dan berdiri. Ruang warna HSV lebih mewakili warna-warna dalam pengelihatannya manusia dibandingkan ruang warna RGB. RGB adalah format ruang warna yang sering digunakan pada citra digital. Ruang warna HSV digunakan untuk mengetahui warna sesungguhnya dari gambar. Proses *thresholding* citra pada penelitian ini digunakan untuk mempermudah perhitungan piksel. Jika kedua pemrosesan citra ini digabungkan, maka dapat digunakan untuk deteksi kondisi sederhana, seperti duduk dan berdiri. Dalam pengujianya deteksi posisi duduk dan berdiri mencapai akurasi 100%. Posisi duduk dan berdiri dapat dideteksi dengan baik jika pengguna sistem melakukan pengaturan kamera menyesuaikan tinggi badannya.
3. Algoritma Haar cascade classifier dapat digunakan untuk mendeteksi wajah dan mata walaupun mata dalam kondisi tertutup. Sistem deteksi perhatian tidak memenuhi perancangan jika wajah dan mata tetap terdeteksi, dengan bantuan konversi Luma pada komponen ruang warna YCbCr setelah konversi RGB ke *Gray* maka kondisi ini dapat diatasi. Haar cascade mendeteksi wajah dan mata dengan membandingkan fitur haar-like pada gambar *gray*. Algoritma ini dapat mendeteksi wajah manusia dengan cukup akurat jika subjek tidak memalingkan wajah dan kepala lebih dari 15°. Pada pengujian akurasi penerapan algoritma Haar Cascade, sistem dapat mendeteksi wajah dengan baik selama subjek dalam jangkauan kamera.

4. Hasil pengujian sistem deteksi perhatian operator kamera pengawas terhadap monitor secara keseluruhan mendapat hasil yang cukup baik dengan akurasi deteksi duduk dan berdiri sebanyak 100% dan akurasi hasil klasifikasi sistem saat mendeteksi perhatian subjek terhadap monitor sebanyak 98.33%. Dimana jika subjek dikategorikan tidak memperhatikan monitor menurut manusia, maka sistem juga dapat mengkategorikan subjek dengan baik.

7.2 Saran

Saran yang diberikan untuk sistem ini guna penelitian yang lebih lanjut kedepannya berupa:

1. Deteksi duduk dan berdiri subjek hendaknya dikembangkan lagi dengan metode yang lebih akurat tanpa pengaturan posisi kamera.
2. Algoritma yang digunakan untuk mengenali wajah dan mata hendaknya dikembangkan lagi agar lebih akurat
3. Untuk mengenali wajah dan mata manusia sebaiknya menggunakan kamera dengan resolusi lebih baik



Daftar Pustaka

- Bytemark, 2017. *Raspbian*. [Online] Tersedia di: <https://www.raspbian.org/> [Diakses pada 31 Januari 2018].
- Debnath, P. P., Hasan, A. F. M. R. & Das, D., 2017. *Detection and Controlling of Drivers' Visual Focus of Attention*. Bazar, International Conference on Electrical, Computer and Communication Engineering (ECCE).
- Donald, D. C., 2015. *How long can CCTV operators concentrate for?*. [Online] Tersedia di: <http://www.securitysa.com/8289a> [Diakses pada 8 Januari 2018].
- Farnell, 2017. *Specifcations Buzzer*. [Online] Tersedia di: www.farnell.com/datasheets/2171929.pdf [Diakses pada 17 April 2018].
- Giannakakis, G., Manousos, D., Chaniotakis, V. & Tsiknakis, M., 2018. *Evaluation of head pose features for stress detection and classification*. Las Vegas, International Conference on Biomedical & Health Informatics (BHI).
- Kolkur, S. et al., 2016 . Human Skin Detection Using RGB, HSV and YCbCr Color Models. *ICCASP/ICMMD Advances in Intelligent Systems Research*.
- Krishna, M. & A. Srinivasulu, 2012. Face Detection System On AdaBoost Algorithm Using Haar Classifiers. *International Journal of Modern Engineering Research (IJMER)*.
- Logitech, 2018. *Logitech HD WEBCAM C270*. [Online] Tersedia di: <https://www.logitech.com/en-gb/product/hd-webcam-c270#specification-tabular> [Diakses pada 26 Maret 2018].
- OpenCV, t., 2018. *OpenCV*. [Online] Tersedia di: <https://opencv.org/> [Diakses pada 10 January 2018].
- Putro, M. D., Adjij, T. B. & Winduratna, B., 2012. Sistem Deteksi Wajah dengan Menggunakan Metode Viola-Jones. *SciETec Science, Engineering and Technology*.
- Qiming, L. et al., 2017. *The design of intelligent crowd attention detection system based on face detection technology*. Yangzhou, International Conference on Electronic Measurement & Instruments.
- Rahman, M. A., Purnama, I. K. E. & Purnomo, M. H., 2014. Simple Method of Human Skin Detection using HSV and YCbCr Color Spaces. *IEEE*, pp. 19-21.
- Raspberry Pi, F., 2016. *RASPBERRY PI 3 MODEL B*. [Online] Tersedia di: <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/> [Diakses pada 29 Januari 2018].
- RaspberryPi, 2016. *Compute Module Datasheet*, s.l.: Raspberry Pi (Trading) Ltd..
- RaspberryPi, 2016. *Schematics*. [Online] Tersedia di: https://www.raspberrypi.org/documentation/hardware/raspberrypi/schematics/r_pi_SCH_3b_1p2_reduced.pdf [Diakses pada 22 May 2018].

- Rybak, N., Hassall, M., Parsa, K. & Angus, D. J., 2017. New real-time methods for operator situational awareness retrieval and higher process safety in the control room. *IEEE*.
- Santoso, H. & Harjoko, A., 2013. HAAR CASCADE CLASSIFIER DAN ALGORITMA ADABOOST UNTUK DETEKSI BANYAK WAJAH DALAM RUANG KELAS. *JURNAL TEKNOLOGI*.
- Semiun, Y., 2006. *Kesehatan Mental*. 3 ed. Yogyakarta: Kanisius.
- Setiawan, E., 2012. *Kamus Besar Bahasa Indonesia (KBBI)*. [Online] Tersedia di: <https://www.kbbi.web.id/fokus> [Diakses pada 5 Juni 2018].
- Shirbhate, N. & Talele, K., 2016. Human Body Language Understanding for Action Detection using Geometric Features. *IEEE*, p. 603.
- Usuman, I., Dharmawan, A. & Frisky, A. Z. K., 2012. Sistem Pendeteksi Kulit Manusia Menggunakan Segmentasi Warna Kulit Pada Tipe Citra HSV (Hue Saturation Value). *IJEIS*.
- Vala, H. J. & Baxi, A., 2013. A Review on Otsu Image Segmentation. *International Journal of Advanced Research in Computer Engineering & Technology*.
- Viola, P. & Jones, M., 2001. Rapid Object Detection using a Boosted Cascade of Simple Features. *IEEE*.
- Zhang, T., Hu, H.-M. & Li, d. B., 2018. A Naturalness Preserved Fast Dehazing Algorithm Using HSV Color Space. *IEEE*.